



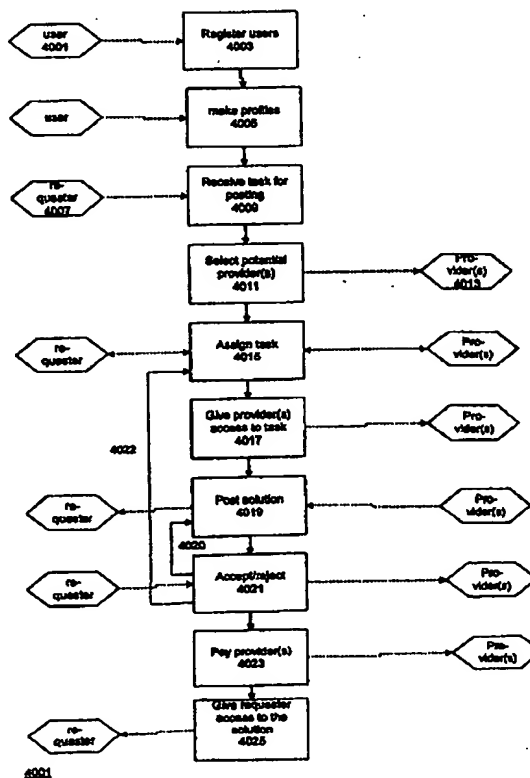
INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification ⁷ : G06F 17/30		A1	(11) International Publication Number: WO 00/29989
			(43) International Publication Date: 25 May 2000 (25.05.00)
(21) International Application Number: PCT/US99/27270		(81) Designated States: JP, US, European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).	
(22) International Filing Date: 18 November 1999 (18.11.99)			
(30) Priority Data: 60/108,834 18 November 1998 (18.11.98) US		Published <i>With international search report. Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.</i>	
(71) Applicant (for all designated States except US): HOTDISPATCH, INC. [US/US]; P.O. Box 425645, Cambridge, MA 02142 (US).			
(72) Inventors; and (75) Inventors/Applicants (for US only): SAYED, Hazem [US/US]; HotDispatch, Inc., P.O. Box 425645, Cambridge, MA 02142 (US). BLUMBERG, Andrew, J. [US/US]; HotDispatch, Inc., P.O. Box 425645, Cambridge, MA 02142 (US).			
(74) Agent: NELSON, Gordon, E.; 57 Central Street, P.O. Box 782, Rowley, MA 01969 (US).			

(54) Title: **SYSTEM FOR ELECTRONIC COMMERCE IN NON-STANDARDIZED SERVICES**

(57) Abstract

A system is provided for electronic commerce in non-standardized services. Users (4001) register with the service dispatch system and make one or more profiles (4005). Service requesters (4007) send a task to the service dispatch system for posting. At (4011), the service dispatch system selects potential service providers (4013) for the task based on the classification of expertise indicated by the users when they submitted their profiles and the ratings of the users to select one or more service providers who are qualified to perform the task. Dispatch system then pays service provider (4013) for delivering the particular service or task to requester (4007).



BEST AVAILABLE COPY

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece	ML	Mali	TR	Turkey
BG	Bulgaria	HU	Hungary	MN	Mongolia	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MR	Mauritania	UA	Ukraine
BR	Brazil	IL	Israel	MW	Malawi	UG	Uganda
BY	Belarus	IS	Iceland	MX	Mexico	US	United States of America
CA	Canada	IT	Italy	NE	Niger	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NL	Netherlands	VN	Viet Nam
CG	Congo	KE	Kenya	NO	Norway	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NZ	New Zealand	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	PL	Poland		
CM	Cameroon	KR	Republic of Korea	PT	Portugal		
CN	China	KZ	Kazakhstan	RO	Romania		
CU	Cuba	LC	Saint Lucia	RU	Russian Federation		
CZ	Czech Republic	LI	Liechtenstein	SD	Sudan		
DE	Germany	LK	Sri Lanka	SE	Sweden		
DK	Denmark	LR	Liberia	SG	Singapore		
EE	Estonia						

System for electronic commerce in non-standardized services

Cross references to related applications

The present patent application claims priority from United States provisional application
5 60/108,834, H. Sayed et al., *Market system for Web-deliverable services*, filed 11/18/98.

Background of the invention

1. Field of the invention

The invention relates to systems for electronic commerce generally and more specifically to
10 systems for electronic commerce in services.

2. Description of related art

The development of the Internet and more particularly of the World Wide Web have led to the
explosive growth of electronic commerce, or *e-commerce*. The World Wide Web has made
15 Web sites directly accessible from anywhere on earth, and has in the process so expanded
possible markets and reduced transaction costs that it has opened a whole range of new
possibilities for both very large and very small enterprises. The business of book selling can
serve as an example: on the one hand, amazon.com® has in a few years become one of the
giants of the book selling business; on the other hand, small dealers in specialized or old books
20 are now easily accessible from anywhere in the world.

At present, the advantages of e-commerce in increased access of vendors to customers of
customers to vendors and in reduced transaction costs are being reaped only with regard to
goods and standardized services. As one would expect, people are most comfortable with e-
25 commerce where there is no need to see the goods or try them out before purchasing them, as
is the case for example with personal computers, new books, or shares of stock. E-commerce
in services has been limited to services that are so standardized that they can be represented by
a ticket, a subscription, or a reservation. Examples are e-commerce in theater and concert
tickets, information feeds, hotel reservations, or airline tickets.

30 The increased access to customers and vendors and reduced transaction costs offered by e-
commerce are potentially even more valuable with regard to *non-standardized* services than
they are with regard to goods and standardized services. Examples of non-standardized

services are translating a document, transcribing a medical report, solving a mathematical equation, writing a computer program, writing a report, or doing legal or business research. As can be seen from the examples, non-standardized services have the following characteristics:

- because they are not standardized, they cannot be *reified*, i.e., represented by a ticket, reservation, subscription, or the like; and
- because the providers of the non-standardized services are generally individuals or small businesses, there is no effective market mechanism for bringing purchasers and providers together or for determining prices.

Often, the providers work on a part-time basis. For example, foreign engineering graduate students in the United States have long supplemented their income by doing technical translation in their spare time. Similarly, lawyers who are staying home with their children often do legal research for other lawyers.

In many cases, non-standardized services also have the following characteristics:

- special training or experience is needed to do the job; and
- the cost of the job depends on factors such as the job's length, its difficulty, and the time the service provider is given to do the job.

For instance, a translation of a French newspaper article will generally cost less than a translation of a French technical paper of the same length and the price will rise steeply as the period for completing the work gets shorter.

From the point of view of a customer for such a non-standardized service, the first problem is to find someone to do it at all. The more specialized the service is, the less likely the customer is to know someone who provides it or to be able to judge whether the person the customer has found is in fact capable of providing the service. Once the person has been found, the next problem is setting a price: since there is effectively no market, the price generally ends up being unfair to one or another of the parties. Then there are all of the problems of dealing with small businesses or individuals arise: typically, they don't take credit cards and communication may be difficult or sporadic at best. Finally, if the job is not finished, is finished late, or is otherwise not properly done, the customer often has no practical recourse.

The provider of a non-standardized service has the reverse problems, and the smaller the provider is, the more difficult the problems are. The first problem is of course finding the

customer. Advertising is expensive, and if potential customers are widely dispersed, there is often no cost-effective way of doing it. Once the customer has been found, setting a price in a vacuum is as hard for the provider as it is for the customer. The provider has little or no information about the customer's ability to pay, and once the customer has the deliverable, be it a legal opinion or a translation, the customer has little or no incentive to pay. When the customer does not pay, the provider, too, often has no practical recourse, since the size of the transaction often cannot justify legal action to compel payment.

The high transaction cost of locating qualified providers of non-standardized services has prevented some things from being done at all. For example, a manufacturer of a specialized product may want some feedback from potential users of the product; given the cost of locating the potential users, getting into contact with them, and finding out what their needs are, the manufacturer often simply decides to rely on what the salespeople think. On the other hand, if some way could be found to make contact easy, many potential users would be glad to provide the feedback, particularly if they got some compensation for the time they spent doing it. The high transaction cost of locating customers has also discouraged people such as stay-at-home spouses, students, and consultants from providing non-standardized services.

What is needed if the benefits of e-commerce are to be extended to commerce in non-standardized services is solutions to problems such as the following:

- making customers for such services and providers of such services more accessible to each other;
- vouching for the credentials of the service provider and the quality and reliability of the service he or she provides;
- determining what the price for the service should be;
- vouching for the ability and willingness of the customer to pay;
- keeping track of the status of the job;
- providing for communication between the customer and the service provider;
- giving the customer enough access to the deliverable to decide whether the job was well done but giving him complete access only after he has paid;
- providing audit trails in case of disagreement between the customer and service provider; and
- providing low-cost dispute resolution facilities for settling such disagreements.

It is an object of the present invention to provide solutions to these and other problems of expanding e-commerce to include non-standardized services.

Summary of the invention

5 The invention attains its object by providing a service dispatch system that mediates between a service requester and a set of service providers. The service dispatch system mediates with regard to the identities of requester and providers, with regard to assignment of a task requested by a service requester to a service provider, with regard to negotiation of a price for the task, with regard to acceptance of a solution for a task by the service requester, with regard
10 to payment of the price, and with regard to dispute resolution between the parties. Other embodiments may not mediate in all of these areas. The service requester and the service providers have access to the service dispatch system by an interactive interface. In a preferred embodiment, service requesters and service providers interact with the service dispatch system via the Internet.

15 The service dispatch system receives a task and a specification of a price from the service requester. The service dispatch system responds to this task posting by assigning one or more service providers to the task. When a service provider has a solution for the task, he or she posts the solution to the service dispatch system, which in turn indicates to the service
20 requester that the solution is available. When the service requester indicates to the service dispatch system that he or she accepts the solution, the service dispatch system pays the service provider the price and provides the solution to the service requester. The service dispatch system thus makes it possible for service requesters and service providers to easily locate each other and ensures that the service requester receives the solution and that the service provider
25 receives the price.

The service dispatch system of the invention further permits users to define one or more profiles for themselves. Each of a user's profiles presents a different view of the user to other users of the system. For example, a user's profile may indicate the expertise and education that
30 the user has for solving a particular class of task. Profiles also permit a service requester to post a task anonymously and a service provider to disclose only as much personal information as is necessary to show competence in solving tasks of a particular class. When the service dispatch system assigns a task, it assigns it to one or more profiles. When the service

requester pays for a solution, the service dispatch system obtains rating information from both the service requester and service provider. The cumulative ratings for a profile are part of the profile and may be made available to service requesters or providers. The service dispatch system thus provides a market for service requesters and providers which provides service
5 quality information, but otherwise provides as much anonymity as is desired between the participants and is therefore less affected by the prejudices of either the service requester or the service provider than traditional markets.

When a service provider sets up a profile, he or she indicates his or her areas of expertise, and
10 when a service requester posts a task, he or she classifies the task. The service dispatch system uses the information that it has about the classification of the task and the ratings of the profiles to select a number of service providers to whom the task might be assigned. In some cases, the service dispatch system will automatically assign the task to the selected service providers; in others, it may provide a list of selected profiles to the service requester, with the final
15 selection being made by the service requester. As will be explained below, the selection may be made on the basis of price.

The service dispatch system permits a service requester to establish a method of determining the price of the service. In a preferred embodiment, establishment of the price may involve
20 bargaining between the service requester and the service provider or a bidding system. In both cases, the service dispatch system mediates the bargaining and the bidding.

When a service provider posts a solution, the service requester may either accept the solution, completely reject the solution, or request that it be modified. Where possible, the service
25 dispatch system will provide the requester with enough information about the solution to permit an informed judgment of the solution's quality, but not with the entire solution until the requester has accepted and paid for it.

The service dispatch system also mediates between service requester and service provider with
30 regard to payment. Each profile may include specifications of how the profile will pay if it is functioning as a service requester and how it is to be paid if it is functioning as a service provider. On acceptance of a solution by a requester, the service dispatch system transfers payment from the requester to the provider.

The service dispatch system maintains enough information about the interactions between service requesters and service providers and about rejected and accepted solutions to permit a complete audit trail to be made of a transaction between a service requester and a service provider. Additionally, one of the services provided by the service dispatch system is arbitration of a disputed transaction on the basis of the audit trail.

Other objects and advantages will be apparent to those skilled in the arts to which the invention pertains upon perusal of the following *Detailed Description* and drawing, wherein:

10

Brief description of the drawing

FIG. 1 is an overview of a system for handling requests for non-standardized services;

FIG. 2 shows a first portion of the database used in a preferred embodiment of the system of **FIG. 1**;

15 **FIG. 3** shows a second portion of the database used in a preferred embodiment of the system of **FIG. 1**;

FIG. 4 shows a third portion of the database used in a preferred embodiment of the system of **FIG. 1**;

FIG. 5 shows the SQL definition of account record 205;

20 **FIG. 6** shows the SQL definitions of education record 217 and expertise record 221;

FIG. 7 shows the SQL definition of pay method records 209 and 213;

FIG. 8 shows the SQL definitions of profile record 225 and rating records 229, 233, and 237;

FIG. 9 shows the SQL definitions of task record 249 and solution record 269;

25 **FIG. 10** shows the SQL definitions of pricing scheme record 307, negotiation state record 311, and negotiation history record 265;

FIG. 11 shows the SQL definitions of provider state record 261 and activity record 257;

FIG. 12 shows the SQL definitions of status history record 253, solution history record 273, and message record 409;

FIG. 13 is an overview of database manipulation routines 115;

30 **FIG. 14** is the initial Web page of the user interface;

FIG. 15 shows a first part of the user interface for setting up an account;

FIG. 16 shows a second part of the user interface for setting up an account;

FIG. 17 shows a second part of the user interface for posting a question;

- FIG. 18** shows a first part of the user interface for posting a question;
FIG. 19 shows the user interface for viewing a solution;
FIG. 20 shows the user interface for rating a service provider;
FIG. 21 shows the user interface for allocating payments among service providers;
5 **FIG. 22** shows the user interface for viewing messages;
FIG. 23 shows the user interface for posting a project;
FIG. 24 shows the user interface for selecting a pricing scheme for a task;
FIG. 25 shows the user interfaces for categorizing a task and for setting up sealed bidding for
the task;
10 **FIG. 26** shows the user interface for viewing open tasks;
FIG. 27 shows the user interface for finding service providers;
FIG. 28 shows the user interface whereby a service provider may view posted tasks;
FIG. 29 shows the user interface whereby a user can view messages concerning tasks;
FIG. 30 shows the user interface whereby a user can view tasks;
15 **FIG. 31** shows the user interface whereby a user can edit his or her account;
FIG. 32 shows the user interface whereby a user can edit his or her profiles;
FIG. 33 shows the user interface whereby a user can view his or her balances in system 101;
FIG. 34 shows the user interface whereby a user can view task messages for tasks involving
one of his or her profiles;
20 **FIG. 35** shows the user interface whereby a service requester can post a task for assignment;
FIG. 36 shows the user interface whereby a service provider can offer to perform a task;
FIG. 37 shows the user interface whereby a service requester receives an offer from a service
provider;
FIG. 38 shows the user interface whereby a service requester accepts an offer from a service
25 provider;
FIG. 39 shows the user interface whereby a service provider accepts an assignment of a task;
and
FIG. 40 is a flowchart of the operation of system 101.
- 30 Reference numbers in the drawing have three or more digits: the two right-hand digits are
reference numbers in the drawing indicated by the remaining digits. Thus, an item with the
reference number 203 first appears as item 203 in FIG. 2.

Detailed Description

The following *Detailed Description* will begin with an introduction explaining the utility of a system for e-commerce in non-standardized services, termed herein a *service dispatch* system, and providing an overview of its operation, will then present an overview of a preferred embodiment, and will finally describe details of a preferred embodiment, including the structure of the database used therein and the user interface for the system.

Introduction to the service dispatch system

The service dispatch system of the invention is the first-of-its-kind mechanism for transacting "web-deliverable" services on the Internet. The service dispatch system is designed to provide a complete environment for matching a given service task to a qualified service provider, and tracking the transaction from the posting of the task through the payment for services rendered. In between, the service dispatch system manages all necessary interactions between the parties involved in the exchange. A *task* is to be understood here as anything in a range from an answer to a question through a large software engineering or research project. In a preferred embodiment, there are two subtypes of tasks: *questions*, where all that is required is that a service provider who knows the answer provide it to the service requester, and *projects*, where the service provider must undertake considerable independent effort to provide the requested service.

The service dispatch system provides a mechanism to out-source tasks via the Internet. Whether the task is the translation of a document, transcription of a medical report, solving a mathematical equation, getting a legal opinion, or a consumer opinion, the service dispatch system will find one or more people with the expertise, get them working on it and send back the solution for acceptance. The service dispatch system matches talent with problems and provides a medium for both parties to transact their business.

E-commerce solutions and services have to date been primarily directed to the sale of tangible goods (such as books, computers, cars, etc.) and secondarily directed to branded services (such as airline tickets, hotel reservations, newsfeeds) which typically have physical realizations (the flight, the hotel, and so forth). Mechanisms typically focus on negotiating a price agreement; the services provided are generally fixed and the buyer finds the services by external measures. Examples include the fixed price of a company store (e.g. dell.com), the auction model of

ebay.com or the reverse auction model of priceline.com. The one paradigm involving non-standardized services is an old one; the classified pages brought straight into the electronic age.

5 The service dispatch system is qualitatively different, in that it is intended to provide the substrate for a free market for any services that are deliverable via the Web (specifically, are reducible to text, images, audio, video, or other computer formats). Obvious possibilities include (but are not limited to) photographs, recordings, graphic design projects, banner advertising design, document translation, book reviews, mathematical problem solutions, programming tasks, medical transcriptions, and so forth. It should be noted, however, that the
10 task management features of the service dispatch system are also usable for services that are not deliverable via the Web.

The service dispatch system is not just another classified service. Unlike a classified page, the service dispatch system provides specific mechanisms for linking service providers and
15 service buyers and managing all of their transactions in a secure fashion. It provides a complete market, in which both parties can have the security provided by a well-defined process for the exchange of services.

The service dispatch system provides the security of a full audit trail and payment typically via
20 zero-knowledge or "ambiguated" solution interaction (for example, a company might buy photographs based on low resolution previews). Moreover, full arbitration is available in all cases (and all necessary information for arbitration is automatically logged, making complaint resolution easy and fair).

25 Furthermore, the service dispatch system permits both parties to be partially or fully anonymous. As such, it is well-suited to facilitate the exchange of services based on skill alone, without reference to other prejudices. A company buying a banner ad graphic does not need to know the race, sex, age, or anything else about the person who created the ad, as long as it looks good. The service dispatch system makes this notion a reality.

30 The service dispatch system is designed to reduce the granularity level of service exchange; it is intended to facilitate the transfer of services which are too small to be handled by

contemporary means (i.e. hiring a new employee), in a way which is much faster and more reliable than posting a classified ad.

5 For small, irregular tasks, the service dispatch system is designed to provide a cheaper and easier solution than traditional contract work; it is specifically targeted at helping small businesses with limited funds meet their needs for specific services in a timely and efficient fashion.

10 Moreover, the service dispatch system is designed to permit the utilization of the skills of a vast body of workers which are underutilized in today's economy. Workers who cannot take a regular job, for a variety of reasons, can use the service dispatch system to sell their special expertise on a time available basis. For example, the service dispatch system is ideally targeted at students --- a highly skilled population which is inaccessible due to time constraints. The service dispatch system is also ideal for at-home parents and part-time workers; in every
15 case, the structure and format of the exchange permit the sale of skills in a fluid, flexible way consistent with the outside constraints.

Finally, the service dispatch system creates a truly global and 7x24 hour market for services, allowing service buyers to utilize talent outside their geography and allowing talent to find
20 markets outside its local economy.

Examples of services which may be provided via the service dispatch system

1. Translation Tasks:

25 You have a document you want translated from English to French. You can send the document to the service dispatch system in a number of ways. If it's a text or .pdf file, just upload it via the web. If it's a handwritten document, you can mail it and the service dispatch system will scan and upload it.

2. Transcription Tasks:

30 You have an audio tape of a doctor's notes or handwritten notes you want transcribed. Send the tape or the paper and the service dispatch system will convert it to digital form and upload it. You can ask the service dispatch system to only show that task to qualified service providers (medical transcribers) and to show you a list of the one's

interested/available for you to assign the task to. Once assigned, the service provider will work on the task and upload their result.

3. Legal Opinion Tasks:

- 5 You want to know your rights as a tenant in a given town. Submit a legal opinion task describing your situation. The service dispatch system will show it to lawyers with that expertise and assign it to one of them. Their response will include references to check their credentials.

10 4. Problem Tasks:

You are working on a dissertation and need a solution to a complex mathematical formula. Submit a mathematical problem task and request that it be shown as widely as possible and that you will pay for the result. Mathematicians and students who are users of the service dispatch system will see it, and the first to solve it will get the task.

15

5. Opinion Tasks:

- 20 You want to conduct a survey of what people think about a political issue. Submit an opinion task with your questions (using the service dispatch system's survey tool) and specify how many people you want the opinions of and how much you are paying for each opinion. The service dispatch system will display your task, provide them with an interface for taking answering your questions and send the answers to you in both full and summary forms.

- 25 You want to know if people in a particular country think you would look better with one hair style or another. Post that question on the service dispatch system along with the different photographs, and specify how many respondents you want and from what country. The service dispatch system will provide you the answers.

6. Photography Tasks:

- 30 You need a photograph of a sunset over the Serengeti. You post it as an open to the public task.

7. Audio Tasks:

You need a recording of the sound of a koala bear. You post it as an open to the public task.

8. Graphic Tasks:

5 Banner ad design.

Drafting services. Service buyer submits scanned sketch, service provider returns drawing.

9. Programming Tasks:

You need a banner ad for your company...

10

10. Reward Tasks:

Have you seen this man, he's wanted by the FBI...

Have you seen this child, he's been missing for..

15 11. Competition Tasks:

Design entries are invited for a new garden at the corner of Prospect and Mass Ave..

Entries accepted until midnight December 31. Winners will be announced by March 1.

First prize will receive

20 12. Research Task:

Legal research. Service buyer is a lawyer, service provider is a paralegal.

Reference library services, trademark searches, etc.

13. Marketing Research Tasks:

25 Will pay \$x for each opinion on this product.

14. Time Sharing Tasks:

Will pay \$/hour for compute time to run this applet.

30 15. Programming Task:

Need JavaScript code that will pop-up a window when...

16. Book Review Task:

Need a 1000 word review of the new "Internet for Dummies" books.

17. Movie Review Tasks:

Launching a new movie site. Looking for 100 word reviews of current releases.

5

18. Tax preparation Tasks:

19. Advertising via survey tasks:

10 Have a single question survey asking opinion of the enclosed banner ad. Service provider (the survey taker) paid viewing ad and clicking "like" or "don't like" button. Service buyer (advertiser) guaranteed viewership of their message and feedback on its reception. Being paid to view advertising.

20. Arbitration Tasks:

15 Arbitration service for small disputes.

Functional description of the service dispatch system

The functions performed by the service dispatch system are the following :

1. User account creation and login
- 20 2. Task posting
3. Task assignment and negotiation
4. Task solution submission
5. Task solution review and acceptance
6. Payment
- 25 7. Task solution delivery
8. Dispute arbitration
9. Communications management
10. Privacy and security

30 1. User account creation and login

a. user account management

User login is required for service requesters and optional for service providers (although refusal to login may result in circumscribed access). The system permits the creation of a user account for authentication purposes which may contain one or more service requester and/or

service provider profiles. The profiles are mechanisms which permit users of the system to present different aspects of themselves to it. When a user account is created, the system will prompt the user for a username and password which are stored in a persistent database and thereafter used for login authentication. For a given session, the system will allow the user (having created an account in the past) to login as either a service requester or a service provider. Prior to entry to the system, the user must enter the username and password for a valid account in the persistent database.

b. service provider profile creation

The user can create one or more service provider profiles via a Web form. Each profile collects information including at least a handle, an e-mail address for correspondence, payment information, and information regarding the kinds of service they wish to provide. The profile may optionally collect further information such as name, address, description, and so forth. To be eligible for certain kinds of tasks, evidence of certification may be collected as well (see certification management below). Profile information is stored in a persistent database and can be edited at any time by an authenticated user.

c. service requester profile creation

The user can create one or more service requester profiles via a web form. Each profile collects information including at least a business name, business address and phone, an e-mail address for correspondence, and payment/billing information. The profile may optionally collect further information such as a description. To be eligible to post certain kinds of tasks, evidence of certification may be collected as well (see certification management below). Profile information is stored in a persistent database and can be edited at any time by an authenticated user.

d. certification management

In order to be eligible for certain classes of tasks (either for service buying or service provision), outside certification (e.g. license to practice law) may be collected. Such certification is collected in the form of text information (i.e. license number) and/or uploaded files. Before certification is granted, various off-line processes may occur (i.e. a phone call to

the relevant registry). All information is stored in the persistent database and may be edited at any time by an authenticated user.

2. Task posting

5 A task can be posted by any registered user with a service requester profile. The posting of the tasks involves the collection of at least the following pieces of information (if not explicitly provided then default values will be used) :

- a. task category (translation, transcription, photography, and so forth)
- b. task title
- 10 c. short description
- d. task body
- e. display options (public, public to qualified providers, private, etc.)
- f. solution options
 1. number of solutions solicited
 - 15 2. number of solutions actually purchased
 3. method in which purchased solutions will be selected (i.e. auction, competition, first-in)
- g. pricing information (maximum price, whether bargaining is acceptable, etc.)
- h. assignment information (whether the task can be automatically assigned, methods in
20 which assignment is confirmed, etc. [see Task assignment below])
- i. time constraints (for various phases of task resolution)

All of the task information is stored in the persistent database. However, the task can be edited only prior to assignment (see below) by an authenticated user.

25 3. Task assignment

The system permits three possible methods of task assignment, automatic assignment, manual assignment, and posting for solicitation. The task may be switched between methods at any time (although resulting behavior may depend on previous assignments made).

30 In automatic assignment, a number of service providers (the number depending on the number of solicited solutions above and possibly other information) are chosen based on criteria including time in the system, performance history, and stated category interests. These service providers are notified of the interest of the buyer in assigning the task to them.

In manual assignment, the buyer is provided with the opportunity to browse the space of service providers to select some number to be notified of opportunity of assignment. The browsing process involves some view of the total space of service providers, constrained by such factors as interest, time in the system, performance history, and possibly other things.

In posting for solicitation, the task description is made accessible to a relevant community of service providers and assignment requests are solicited.

10 All assignment decisions are logged in the persistent database and cannot be edited by the authenticated user (they are stored for arbitration purposes).

In some cases, assignment decisions will be made on the basis of price negotiations. A preferred embodiment of the system supports direct negotiations between the service requester and interested service providers and also supports bidding systems, including reverse auction, sealed bidding, and Dutch auction.

Notification of service requesters involves a message (sent through the internal messaging system and/or e-mail) stating the interest of the service provider as well as providing some amount of information about the task. Depending on privacy settings, varying degrees of information about the service requester will be provided. The information about the task includes at least a short description, information about price, information about acceptance criterion, and information about the assignment process (such as how many others were notified, and so forth).

25 A notified service requester may respond to such a message in one of four possible ways.

1. The service provider may accept the assignment request.
2. The service provider may conditionally accept and make a counteroffer regarding terms of the assignment.
3. The service provider may reject the assignment request.
4. The service provider may ask for clarification regarding the assignment request.

The service requester will be notified of all such response decision.

In response to information from a service provider, the service requester may respond as follows :

- 5 • If the service provider has accepted the assignment request, the service requester can either confirm the assignment or reject the service provider according to the protocols specified in the task description.
- If the service provider has rejected the assignment request, no response is permitted.
- If the service provider has conditionally accepted and made a counteroffer, the service
10 requester may accept the offer (which has the same effect as if the service provider had accepted the assignment request, except some piece of the task description has been modified), may respond with a counteroffer (which permits a response from the service provider as above), or may reject the service provider.

The cycle of counteroffers will continue until a rejection occurs or an agreement is reached.

- 15 This cycle may be automated and coordinated between assignment requests (i.e. in an auction model).

Notification and negotiation of assignment requests can continue until a total number of assignments specified (as solutions solicited) in the task description are confirmed. Once a
20 service provider has been assigned a task, the service provider is then given access to the complete task body in the event that the task body was not made previously available. The access to the complete task body gives the service provider the information needed to do the task.

25 4. Task solution submission:

Task solutions can be submitted in one of three ways.

1. A registered user possessing a service provider profile which has been assigned the task may submit a solution.
2. A registered user possessing a service provider profile may submit a solution to a
30 viewable task.
3. A registered user without a service provider profile or a unregistered user may submit a solution to a viewable task (in the case of the unregistered user, this is public tasks only).

Solution submission in the latter two cases is only possible if the task has not been assigned to any service providers.

When the service provider has completed the solution, the system permits the service provider to upload the solution as well as associated information, including at least either an "ambiguation method" or an uploaded ambiguated version of the task.

Ambiguation is a process whereby the solution is "blurred" in some way so that (ideally) the buyer can verify the necessary features of the solution but does not actually have possession of the solution. Examples might include (but are not limited to) actual blurring of photographs (or provision of thumbnails), display of fragments of text documents or audio files, and so forth. The system will attempt to make provisions so that a service requester cannot assemble a complete solution by examining a number of different submitted solutions. If the service provider is unhappy with the default methods for the given task category, he or she may submit their own "ambiguated" version.

Initially, all submitted information is placed in a temporary holding area (which may be logical or physical) in the persistent database and remains there until confirmation, at which time the ambiguated version of the solution (if it exists) and possibly other information is transmitted to an area accessible to the service requester.

Submission of a solution is only possible as long as the task is "open", meaning that the provider has not already accepted as many solutions as specified in the task description.

5. Task solution review and acceptance.

When a service requester is informed that a solution is ready for his or her review, the system then permits the service requester to inspect aspects of the submitted solution (i.e. the ambiguated version) for acceptance decisions (and may also notify the service requester of any time constraints as specified in the task posting process for the completion or methodology of the acceptance process).

The service requester can respond to the request in one of three ways.

1. The service requester can accept the solution.

2. The service requester can reject the solution outright.

3. The service requester can reject the solution and request changes.

The distinction between cases 2 and 3 is the following. An outright rejection means the assignment of the service provider in question is cancelled and he or she can no longer submit solutions for the task. In addition, it may be that further assignment requests can be made (in order to generate enough potential solutions) depending on settings made during the task submission process.

On the other hand, in case 3 the solution is rejected but the assignment remains and the service provider can upload another submission for future consideration (presuming the task has not closed in the meantime).

The service provider in question will be notified by the system of any of these actions, along with information collected by the system regarding the reasoning of the service requester (which may include free text or selection of a checkbox from a list of rejection reasons).

As part of the acceptance process, the service requester rates the expertise and conduct of the service provider in executing the task and the service provider rates the service requester. The ratings are retained in records associated with the task and the system also aggregates the ratings for tasks posted or performed by a profile to produce cumulative average ratings for the profile. The system may display the ratings to the service requester as part of the task assignment process.

6. Payment processing.

The service requester can pay for the task solution via a variety of methods, including but not limited to a credit-card or pre-paid escrow account. The account is typically charged at the point of solution acceptance (where the charge includes both the price of the solution as quoted to the service provider as well as a surcharge for the use of the service dispatch system).

The service provider is paid via a variety of methods, including but not limited to credit-card refund, check, wire transfer, or addition to an escrow account the service provider maintains with the system. The latter method may be done automatically for amounts below a certain threshold. Where solutions are provided by more than one service provider, the system

provides a mechanism which permits the service requester to allocate the price paid among the service providers. The system also provides a mechanism for adding a tip to the payment.

7. Task solution delivery

- 5 When a service requester accepts a solution, the complete and unambiguated solution is made accessible to the service requester for downloading or on-line viewing.

8. Dispute arbitration

- 10 If a service requester rejects a submitted solution, the system will permit the service provider the opportunity to dispute the buyer's rejection. To participate in the system, all individuals must agree to binding arbitration in case of dispute. When a dispute is logged, a task is posted to the system which is accessible only to arbitrators. The mechanisms of the service dispatch system are used to perform the arbitration as described above, where the body of the task is the complete audit trail from the persistent database of all relevant information as well as the complaint text and response text collected by the system from the service provider and service requester involved in the dispute.

- 20 The arbitrator can rule on the basis of information provided, or can request further testimony (subject to limitations on time for resolution) from either party involved. The arbitrators will then render their decisions and then the system will act on the decision. The decisions made can include the following options :

1. In favor of the service provider, in which case the buyer is obligated to buy the solution.
 2. In favor of the service requester, in which case the provider must abide by the rejection.
- 25 The arbitrator may rule that payment for the costs of arbitration be handled by either the service requester or the service provider (depending on the merits of the complaint) or by some combination thereof. Moreover, the service dispatch system may pay for some component of the arbitration under certain circumstances.

9. Communications management

- 30 All notification and communication between parties described above is carried out by an internal messaging system and/ or via e-mail. The internal messaging system is generally a message-driven system, meaning that responses may be generated only according to specific

situations (detailed above) or in response to a previous message. This enables arbitrary amounts of privacy protection.

5 The messaging system also permits the posting of messages to public and semi-public forums (where the system may restrict access to these based on various user and profile criterion and information).

10 Standard e-mail can be used as either a functional clone of the internal message system (i.e. all internal messages are copied to e-mail, and responses are mediated through an internal mail router) and/ or can be used simply as a notification device (for example, to specify that new internal messages have arrived and should be read).

10. Privacy and security

15 All financial transactions conducted over the world-wide web will be secure. Moreover, arbitrary other information distribution tasks may also be encrypted. The use of profiles in the system makes it possible for both the service requester and the service provider to control the extent to which they remain anonymous with regard to the task (although the system must maintain a certain amount of information to guarantee payment).

20 **Overview of operation of the service dispatch system: FIG. 40**

FIG. 40 is a flowchart 4001 of the operation of the service dispatch system. Dotted arrows indicate communications between the system and the user, service requester, or service provider. At operation 4003, a user 4001 registers with the service dispatch system; at 4005 the user makes one or more profiles for him or herself; at 4009, a user 4001 who is functioning at this point as a service requester 4007 sends a task to the service dispatch system for posting. 25 At 4011, the service dispatch system selects potential service providers 4013 for the task. Depending on the situation the service dispatch system may post the task for solution by all users of the system or even for the general public or may provide a group of one or more service providers among whom the service requester can select. Provision of the group is 30 based on input from the service requester. The input may be the profile name of a service provider or a classification of the task. In the latter case, the service dispatch system uses the classification the areas of expertise indicated by the users when they submitted their profiles,

and the ratings of the users to select one or more service providers who are qualified to perform the task. The system then informs the selected providers of the task.

At 4015, the task is assigned to one or more of the selected service providers. Depending on
5 input from the requester, assignment may be done automatically by the service dispatch system, may simply result from an acceptance of the task by a service provider, or may be the result of negotiation, mediated by the system, between the requester and one or more of the providers. The negotiation may be simply a matter of offers and counteroffers or may take the form of bidding by several service providers.

10 When the task has been assigned to one or more service providers, the assigned service providers are given access to the materials provided by the requester for performance of the task, as shown at 4017. In some cases, the task is completely contained in the posting and this step is not necessary.

15 At 4019, one or more of the providers posts a solution to the task and the requester is notified of the fact that the solution has been posted. The requester is given access to the solution (where possible, the system ambiguates the solution) and then decides at operation 421 whether to accept or reject the solution. In either case, the service provider is informed. If the
20 requester wishes to have a different service provider attempt the task, the system returns to assign task operation 4015 to assign a different service provider (arrow 4022). If the requester merely wants the service provider to correct or improve the solution, the provider is so informed and later posts the improved solution, as indicated by arrow 4020.

25 At 4023, the system performs the operation of paying the providers. In general, what will be paid is the price agreed on during assign task operation 4015, but the requester may also provide input specifying how the price is to be allocated among several service providers and also specifying a tip for one or more of the service providers. When the providers have been paid, the requester is given access to the unambiguated solution, as shown at 4025.

30

Overview of a preferred embodiment of the service dispatch system: FIG. 1

FIG. 1 is an overview of a preferred embodiment of service dispatch system 101. Service dispatch system 101 is implemented in a dispatch system server 109 that has an Internet

Protocol address and includes Internet interface 110. Internet interface 110 has a Web server 111 that obeys the HTTP protocols and thus permits Web page viewing and file transfer, an email client 122 that sends and receives email, and a cybercash client 121 which does secure monetary transactions. Dispatch system server 109 is accessible to its users 104 via their Web browsers and Internet 103. The users 104 include two groups: service requesters 105(a..n) and service providers 107(a..n). Of course, a given user 104 of dispatch server 109 may be a service requester 105(i) for one task and a service provider 107(j) for another.

In addition to Web server 111, dispatch system server 109 includes an application server 112 and dispatch system database 125. Application server 112 includes a dispatcher 114 and a number of application server routines. These routines in turn read and write tables in dispatch system database 125. The information in dispatch system database 125 is organized according to the functions performed by dispatch server 109. Thus, user information 127 is information about the service requesters 105 and service providers 107 who use system 109. Task management information 129 is information that is used to manage a task from the time it is posted by a service requester 105(i) until the time all of the service providers 107(j..o) who successfully provided solutions for the task have been paid. Solution information 133 is the solutions; message information 135 contains the messages that flow among the users of system 101 and between system 101 and the users in the course of operation of system 101. Archive information 137, finally, contains persistently stored information about tasks from system DB 125, and thus provides the information necessary to make an audit trail. The numbers in parentheses are the reference numbers for the database tables that implement these classes of information in the preferred embodiment. These tables will be discussed in detail later.

The routines of database manipulation routines 115 may similarly be grouped according to the database tables they manipulate. User routines 116 manipulate user information 127; task routines 117 manipulate task information 129 and task management information 131; solution routines 118 manipulate solution information 133; message routines 119 manipulate message information 135; and archive routines 120, finally, manipulate archive information 138.

A user interacts with service dispatch system 101 as follows: using his or her Web browser, the user specifies the Web address of dispatch system server 109. Web server 111 responds which responds with an initial Web page that contains a menu which permits the user to select

among interactions with system 101; in response to the selection, Web server 111 provides an initial Web page for the desired interaction. For example, if the user is a service requester 105(i) and wishes to view the tasks that are currently being performed for him or her, the initial Web page for viewing tasks will request that the user identify him or herself. When Web server 111 receives the user's identification, it will forward it together with an indication of what is to be done with the identification to dispatcher 114, which will respond to the indication by invoking a view tasks routine in task routines 117 with the user's identification. The view task routine will use tables in task management info 131 and task info 129 to return a list of the tasks that are currently being performed for the user and their status to dispatcher 114, which will in turn pass the list on to Web server 111, which will incorporate the list into a Web page and return the Web page to service requester 105(i). From the Web page with the list, the user can view further Web pages, with Web server 111 and dispatcher 114 operating as just described. For example, the user may desire to see detailed information about a particular task on the list, and can select one task for detailed viewing. At the detail level, the user can, for example, review the bids for the task or if the task has already been given to a service provider 107(j), send a message to service provider 107(j).

Interaction between a user and dispatch system server 109 may always be at the initiative of the user, or dispatch system server 109 may send email to the user to inform the user of significant events. For instance, if a service provider 107(j) has been selected for a task, a routine in task routines 117 may cause email server 122 to send an email message to service provider 107(j) indicating that he or she has been selected and including the URL of the Web page that will contain the relevant information.

25 **Structure of dispatch system database 125: FIGs. 2-4**

System database 125 is a *relational* database, that is, the information in the database is stored tables. Each table is made up of records. All of the records in a given table contain the same field and fields of records in one table relate the records to records in other tables. FIGs. 2-4 show the tables of dispatch system database 125 and the relationships between records in the tables. A single record is shown in each table, but of course, the same relationships may hold for other records. In general, there are two kinds of relationships: a one-to-many relationship, where a record in one table may be related to many records in another table, and a one-to-one relationship, where a record in one table may be related to only one record in another table. In

FIGS. 2-4, these relationships are indicated by arrows, with the direction of the arrow showing the direction of the relationship (if an arrow points from record 205 to record 217, it shows how record 205 is related to record 217) and an annotation on the arrow indicating whether the relationship is 1 to 1 ("1" on the arrow) or 1 to many ("m") on the arrow. Thus, each account record 205 may have a 1-to-1 relationship with a buyer payment method record 209 and a one-to-many relationship with a profile record 225.

Beginning the discussion of the structure with FIG. 2, the tables shown in FIG. 2 contain information about the users of system 101. Each user has an account record 205 in account table 203. A user may be either a service requester 105 or a service provider 107, or both, and the information in or accessible from a given account record 205 will depend on which role the user has. Thus, an account record 205 has a one-to-one relationship with a buyer payment record 209 and/or a provider payment method record 213. These records indicate how the user is to pay if he or she is a service requester and how the user is to be paid if he or she is a service provider. The account record also has one-to-many relationships with education records 217 and expertise records 221. The education records indicate the user's education and the expertise records indicates his or her expertise.

In order to permit the user 104 to control the degree of anonymity that he or she has with regard to various kinds of tasks and to permit the user to be classified in different ways by system 101, the user is represented in the system not just by account record 205, but by at least one profile record 225. Each profile record permits the user to establish a different degree of anonymity, to respond to different classes of tasks, and to provide different displays of his credentials and expertise. A given task is posted by a profile belonging to the service requester, and responses to the task are by profiles belonging to service providers. In the following, therefore, a task is spoken of as being posted by a *user, profile* tuple and responded to by other *user, profile* tuples. The user's account record 205 may be related to many profile records 225; each profile record may be related to single rating records for expertise (229), conduct (233), and as a buyer (235). These ratings indicate how service requesters have in the past regarded the service provider's expertise and conduct in providing the service and how service providers have in the past felt about dealing with the buyer. Of course, a given profile record 225 will be related to a buyer rating record 235 only if the profile has functioned as a

service requester; similarly, the profile record 225 will be related to the expertise and conduct rating records only if the profile has functioned as a service provider.

5 By means of profile-expertise record 241 and profile education record 245, profile record 225 is linked to one or more of the education records 217 and expertise records 221. This arrangement permits service providers 107 to list per-profile sets of information about their expertise and education. For example, a lawyer who also had formal training in foreign languages might have a profile record 225 for legal tasks and one for translation tasks, with the legal profile record being related to education records 217 and expertise records 221 that
10 showed his or her legal training and expertise and the translation profile record being related to education records 217 and expertise records 221 that showed his or her linguistic training and expertise. The aspect of a user which a given profile record represents is termed herein a *profile* for the user. The lawyer thus has a legal profile and a translation profile in system 101.

15 The remainder of the tables (247, 255, and 271) in FIG. 2 are all related to tasks and will be explained in more detail in the following. A given task is related to the user, profile tuple represented by a profile record 225 by a provider state record 261. As would be expected from the fact that a given user may be dealing with a number of tasks of a given kind, the relationship between a user's profile record 225 and the provider-state records is one-to-many.
20 Task records 249 for the tasks themselves are contained in task table 247 and the solutions for the tasks are contained in solution table 267.

Continuing with FIG. 3, FIG. 3 shows the tables that are related to tasks in more detail. There is a single task record 249 for each task. Each task record 249 is related to one or more user, profile tuples. For a given user, profile, task tuple, this relationship is represented by the user, profile tuple's provider state record 261 for the task. Each user, profile tuple thus has a
25 provider state record 261 for each task to which the user-profile tuple is related.

The contents of a task record will vary according to whether it is related to a service requester profile, to one or more service provider profiles, or to both service requester and service
30 provider profiles. With task records related to a service requester profile, the task record has a 1-to-1 relationship with a buyer payment method record 209 for the service requester, a 1-to-1 relationship with a pricing scheme record 309 for the task, and a 1-to-1 relationship with

activity record 257 for the service requester. Pricing scheme record 309 has a 1-to-many relationship with negotiation state record 313. There is a negotiation state record 313 for every service provider with which the service requester is currently negotiating for performance of the task represented by the task record. Each task record 249 may further be related to a number of status history records 253, each one of which keeps track of a change in the task's status. The status history records 253 for a task permit reconstruction of an audit trail for the task.

Each provider state record 259 and each task record 249 may have many-to-one relationships with activity records 257. A given activity record 257 records a single transaction with regard to a given user, profile, task tuple. All of a task's activity records may be located from task record 249, and the activity records for a given user, profile, task tuple may be located from provider state record 261 for the tuple. When the profile to which activity record 257 belongs is functioning as a service provider with regard to the task, activity record 257 further has a 1-to-1 relationship to account record 205 for the service provider and to a provider payment method record 213. The latter record indicates how the provider to whom the profile belongs is to be paid.

Provider state record 261 for a given user, profile, task tuple has a one-to-one relationship with task record 249 for the task and also has a 1-to-1 relationship with negotiation state record 313 for the user-profile-task tuple and a one-to-many relationship with solution record 269. Each solution record 269 represents one solution by the service provider, profile pair of the tuple for the task represented by task record 249. Solution history table 207 contains a record for each of the solutions provided for the task by the service provider-profile pair. There is a one-to-many relationship, finally, between each solution record 269 and a solution file record 315 which contains the pathname for a file that contains part or all of the solution. As with task records, each provider state record 261 may be related to many status history records 253. Each status history record 253 records a change in the status of the user, profile, task tuple represented by provider state record 271. As with tasks, these status history records 253 are used to construct an audit trail.

As is apparent from the above description, task-related tables 301 permit easy location of all task records 249 for each user-profile tuple. Task records 249 for user, profile, task tuples may

be easily located via provider state records for the user, profile, task tuples. Task category table 303, finally, permits location of task records 249 generally by the kind of task.

FIG. 4 shows the tables that contain information about the messages exchanged between users of system 101. Depending on the embodiment and/or the choice of the user, the messages may remain internal to system 101 and be read or written using Web server 111, or dispatcher 114 may use email client 122 to send a message that originated within system 101 to the recipient user via email and a user 104 may use his or her own email to send a message for another user to server 109 and server 109 may respond to that message by sending email to the recipient. Each message has a message record 409 in message table 407. Since messages concern tasks, have senders and /or recipients who are users 104 and may involve other users 104 as well, there may be one-to-many relationship between profile records 225, account records 205, task records 249, and message records 409. Messages may thus be located by user, by user, profile tuple, and by task. Dispatch system server 109 uses table seen records 405 to keep track of whether and when a given user, profile tuple last examined a given task record 249 or message record 409.

Details of tables in database 125: FIGs. 5-12

FIGs. 5-12 show the SQL query language declarations for the records in the chief tables shown in FIGs. 2-4.

Details of account record 205: FIG. 5

FIG. 5 shows an account record 205. Each account record is indexed by the value in the field `hd_account_id`, 501, and when another record has a relationship to a given account record 205, it will include a field with the value of the given account record's field 501. The fields indicated by 503 track modifications of record 205; the fields indicated by 505 store the contact information for the user. Three of these fields are of particular interest: `username` is the name by which the user represented by record 205 is known to system 101; `taxonomic_categories` specifies categories to which the user may belong, for example, translator or lawyer, and permits searching for users by the categories they belong to; `receive_email` is a flag which indicates whether server 109 is to convert messages internal to server 109 to email and send them to the user represented by account record 205.

At 507, `buyer_pay_method_id` and `provider_pay_method_id` are record identifiers for the user's buyer payment method record 209 and/or provider payment method record 213. Which identifiers are actually present will depend on whether the user is a service requester, service provider, or both. The information at 509 is personal information which the user may prefer not to have visible to other users of system 101. The `visible` field indicates which of this information is to be made visible. `status` 510 indicates whether the user represented by the account is active or inactive. `current_profile_id` 511 indicates which of the profile records 225 belonging to account record 205 is presently of interest. At 513 are shown fields that may contain a short question and a short answer. The user provides the question and answer when he or she registers and they are used to validate a user who has forgotten his or her password.

At 515 is shown the field `balance`, which indicates the balance that system 101 currently owes the user 104 or that the user 104 owes system 101. Here and elsewhere in database 125, amounts of money are represented by an integer number of cents. The fields indicated by 517 are time stamp fields that indicate the last modifications that have occurred in those records belonging to the user that have the record names indicated by the time stamp names. Thus, `expertises_time` indicates the most recent alteration of a record belonging to the user in expertise table 219. The times are represented as an integer number of seconds after a starting time fixed by system 101. In a preferred embodiment, they are used to control caching of parts of database 125. The fields at 517 contain information about any escrow account which system 101 maintains for the user. `upload_directory` field 519, finally, is a field that indicates a directory in dispatch system server 109's file system into which files from the user are to be uploaded.

25

Details of education record 217 and expertise record 221: FIG. 6

The fields that are the keys for these records are indicated at 601 and 609; fields 611 and 603 track modifications of the records; owner ID 613 and 605 contain the value of `hd_account_id` field 501 of account record 205 for the user whose expertise is or education is being specified. In general, a field named `owner_id` in a record contains the index of the record to which the record with the field "belongs", i.e., the record from which the record that contains the field is generally located. In record 217, the fields indicated by 615 contain the actual information about the user's education; in record 221, the fields indicated by

30

607 contain the actual information about the expertise. In both cases, a visible flag permits the user to determine whether the educational information or the expertise information is to be visible to other users of the system.

5 Details of provider payment method record 211 and buyer payment method table 209; FIG. 7
 FIG. 7 shows the records for these tables. In the preferred embodiment, these tables are implemented as a single `pay_method` table that contains the records for both tables. Which table a given record belongs to is indicated by the value of `class` flag 705. At 701 is seen the record's key, which is used in account record 205 and profile record 225. The fields at 703
 10 track modifications of the record; the remaining fields, indicated by 707, contain check account and credit card information that the system uses to make and receive payments.

Details of profile record 211 and rating records 229, 233, and 237; FIG. 8

As mentioned above, profiles in the preferred embodiment permit a user to control how he or
 15 she appears to other users of system 101. A given user may have a number of profile records 211, each of which contains information specifying a different aspect of how the user interacts with system 101. For example, a user 104 may (but need not) have one profile as a service requester 105 and another as a service provider 107, and similarly, if a requester requests different kinds of services or a provider provides different kinds of services, he or she may
 20 have a different profile for each kind of service. When a user registers with system 101, the system provides the user with a default profile. The user may edit that profile or create other profiles.

As shown in FIG. 8, profile record 225 has an index at 801 and change tracking information at
 25 803; `owner_id` 805 has as its value `hd_account_id` 501 for account table 205 for the user 104 that profile record 225 belongs to. `name` field 807 is the name for the profile represented by profile record 225. `taxonomic_categories` 809 indicates the kinds of services the profile represented by profile record 225 performs. It is used to locate potential service providers for a posted task. Email fields 811 indicate the email addresses for messages
 30 addressed to this profile; if `inherit_emails` so indicates, the email addresses are those specified in account record 205. As indicated at 812, profile record 225 may have the indexes 701 for two records in the pay method table.

visible 813 indicates what information in profile record 225 is to be visible to other users of system 101; description 815 is a written description of the profile. At 817 is found the information that is used to rate the effectiveness of the profile; the information includes indices 821 to three rating records 229, 233, and 237. At 819, finally, is found a collection of time stamps indicating the earliest task performed by the profile, the most recent task performed by it, and the most recent times at which various tables belonging to the profile were modified.

In a preferred embodiment, the records of rating tables 227, 231, and 235 are all included in a single rating table. The SQL for the records in the rating table is shown at 229, 233, and 237 in FIG. 8. Each record is accessible by an index 821, has modification information 823, a class value 825, which indicates which of the three kinds of ratings the class contains, and then the current rating, which is cumulative. As shown at 825, the rating is specified as a sum of the past ratings, the number of ratings, and the average.

15 Details of task record 249 and task solution record 269: FIG. 9

Fig. 9 shows the SQL declarations for task records 249 and task solution records 269. Beginning with task record 249, there is one of these records for every user-profile pair concerned with the task. Each task record 249 has its own task_id index 901; at 903 is seen the modification information; owner_id 905 has as its value the profile_id for the user-profile pair's profile. Structured_category 907 is a local encasement of the taxonomic information about the task, and is used to search for tasks by categories and together with taxonomic categories 809 in profile record 225 to match tasks and service providers. dispatch_prototype 909 indicates how system 101 is to deal with the task. For example, if the task is a question, system 101 may not actually assigned the task to any service provider, but instead, only post it, record service providers who provide acceptable solutions, and then allocate payment to them.

The fields at 911 describe the task by name, description, and URL and if the task is to be public, that is, available to be solved by anyone who has access to system 101. public_directory specifies that the solution to the task is public, and consequently can be placed in a publicly-accessible directory in system 101. The fields at 913 contain task pricing and payment information; those at 915 contain values which indicate the state of the task. When the task is being performed by a single service provider, the fields indicated by

917 contain information about the provider. Included here are timestamps 919 for alterations of tables containing information about the provider and task.

Continuing with task solution table 269, each such table has its own index 921 and its own
 5 update information 923; task_id 925 has as its value task_id 901 for task record 249
 that represents the task to which the solution belongs. owner_id 927 has as its value
 provider_state_id 1101 for provider state record 261 that represents the user, profile,
 task tuple to which the solution belongs. The fields at 929 contain status information about the
 10 solution; field 931 contains the task's solution when the solution is small enough to fit within
 task solution record 269; when that is not the case, flag 933 indicates that fact, and there will
 be one or more solution file table records 317, each of which has a field that contains the value
 of task_solution_id 921 and indicates one of the files that contains the solution. At 935,
 there are various time stamps: seen_time indicates the time at which the service requester
 last examined the task; the other two fields are time stamps for the task's status history record
 15 253 and solution history record 273.

Details of pricing scheme record 307, negotiation state record 311, and negotiation history
 record 265: FIG. 10

The tables shown in FIG. 10 implement the arrangements in system 101 for pricing a solution.
 20 Pricing scheme record 307 contains the scheme chosen by the service requester for
 determining the service's price. The record is located from pricing_scheme_id in 913
 in the task's record 249, which has as its value a pricing_scheme_id 1001 for a pricing
 scheme record 307. Then come the modification record, a class field 1005 indicating the
 class of pricing schemes that this record belongs to, which in turn indicates what fields will be
 25 relevant, and then come the fields of pricing information 1006, which describe the mechanism
 used to determine the price, the highest bid (if a bidding system is used), the index of account
 record 205 for the successful bidder, and the commission for the providers of system 101.

The current state of the price determination and a history of prior states are represented by
 30 records 311 and 265. Negotiation state record 311 represents the current state. Each user,
 profile, task tuple involved with a task has a record 311 and the tuple's provider state record
 261 includes the index of its record 311 at reference number 1109. The fields of negotiation
 state record 311 include an index 1007, the usual modification information 1009, the class of

the negotiation state record, polarity 1013, which indicates for an auction whether the bids are to move upwards or downwards, the index 1015 of the pricing scheme record 307 to which record 311 belongs, the service provider bid 1017 represented by the negotiation state record, and a time stamp 1019 for the most recent change in the negotiation history records for the negotiation.

Each stage of the negotiation represented by negotiation state record 311 has a negotiation history record 265. The fields of this record include a key 1021, the usual modification information 1023, a field 1025 whose value is the key for the negotiation state record, a field 1027 for the amount reached in the negotiation at the point in the negotiation represented by record 265, and a whocode field 1029 which indicates whether the party making the move in the negotiation represented by the negotiation history record 265 was the service requester or the service provider. Together, records 307, 311, and 265 for the user, profile, task tuples involved with a task provide a complete audit trail of the negotiations which determined the price for which the service provider is to provide the service.

Details of provider state record 261 and activity record 257: FIG. 11

Provider state record 261 relates a user, profile tuple to a single task record 249, and thus represents a user, profile, task tuple in the system. At 1101 is seen the index for the record and at 1103 the modification information. At 1105 are seen fields whose values are the index of the task record 249 and the index of the profile record 225 for the user, profile tuple. status 1107 contains the current status of the user, profile, task tuple represented by the record. In a preferred embodiment, the status depends on whether the user, profile pair of the tuple is functioning as a service requester or a service provider. Where the user, profile pair is a service requester, the statuses are:

- pending
- authorizing
- released
- offers pending
- assigned
- solved
- paid
- expired

Where the user, profile pair is a service provider, the statuses are:

- uninvolved
- assignment offered
- assignment requested
- 5 • assigned
- assignment offer rejected
- assignment request rejected
- solved
- solution released
- 10 • modification requested
- paid
- expired
- solution rejected

The fields indicated at 1109 contain the indices of the negotiation state record 311 for the user, profile, task tuple and of the solution record 269 for a solution provided by the user, profile, task tuple. price_paid field 111 indicates the price paid by the service requester for the service, the fields at 113 contain the expertise rating given the service provider, profile, task tuple, and the field offer 1115 is the most recent offer made by the service requester to the service provider for the task. The timestamps at 1117 include the time at which the solution was seen by the service requester and the last update times for status history record 253 and solution record 269 for the service provider, profile, task tuple. tip_amount field 1119 indicates the amount of any tip paid by the service requester to the service provider.

Activity record 257 maintains payment information for a given user, profile, task tuple. Fields include the record's index 1121, modification information 1123, fields 1124 for indices for the task record 249 and provider state record 261 for the user, profile, task tuple, field 1125 indicating the class of activity record 1125 (indicating how the fields in 1131 are to be interpreted), and field 1127 containing the index of account record 257 for the provider of the service. The fields indicated by 1131, finally, contain payment information for the user, profile, task tuple.

Details of status history record 253, solution history record 273, and message record 409:

FIG. 12

A status history record 253 is produced each time the status of a task changes; records are made both for the task and for the user, profile, task tuple that caused the change. Thus, both task records 249 and provider state records 261 have one-to-many relationships with task records. As can be seen from the foregoing, the sequence of status history records for a task thus provides an audit trail for the task and the user, profile, task tuples involved with the task. The index of the record is in 1201; its revision tracking fields are shown at 1203; `owner_id` 1205 is the index of task record 249 or provider state record 261 to which status history record 253 belongs. `status` 1207 indicates the new status of the task. The values are the same as those for status field 1107 in provider state table 261.

10

A solution history record 273 is produced each time a provider, profile, task tuple provides a new solution for the task. Fields 1209 and 1211 are the record index and change information fields; field contains the index of solution record 269 for the provider, profile, task tuple. `solution_text` 1215 is a copy of `solution_text` field 931 from solution record 269 for the task whose history is being recorded.

15

A message record 409 is made each time a user of system 101 sends a message to another user thereof about a task, and is also made each time system 101 sends a message about a task to a user. Each record 409 has an index 1217 and modification information 1219; field 1221 is the index of task record 249 for the task that the message concerns. `class` field 1223 indicates the class the message belongs to. There are three classes: system message, requester message, provider message. `text` field 1225 contains the text of the message. The fields labeled 1227 indicate the users that are concerned with the message as sender, recipient, or subject of the message. Each of the fields contains the index for the relevant user's account record 205. `seen_time` 1231 indicates the time that the recipient looked at the message, and the fields indicated by 1233 show whether the sender or recipient has deleted the message. As can be seen from the foregoing, message record 409 provides easy retrieval of messages by sender, recipient, subject, or task, and the sequence of messages for a task is part of the task's audit trail.

25
30

Archiving and audit trails in database 125

System 101 archives the current state of a task by setting a field in the task's task record 249 to indicate that the task is archived and then making a copy of the state of the task when

archived. This copy is part of archive information 137. Subsequently (on a nightly or weekly basis, depending on task volume and system load) system 101 moves archived tasks to a replica of task table 247 which is in a different tablespace (done to optimize disk accesses and permit parallel query efficiency). When the archived task record 249 is moved, copies of the database records that are accessible via the task's index are also moved. Because nothing
 5 ever actually leaves the database and every action is recorded by the creation of a record in a table, an audit trail can be made simply by searching database 125 for the relevant information.

Overview of database manipulation routines 115: FIG. 13

10 FIG. 13 shows the routines used in a preferred embodiment to manipulate dispatch system database 1255. The routines are subdivided into the groups shown in FIG. 1. The following discussion will explain each routine's function and the principle tables in database 125 which are read or written by the routine.

15 User routines 116

- User registration routine 1301 makes an account record 205 for the user and if the user provides profile information, a profile record 225. If the user so indicates, routine 1301 will also make one or more education records 217, expertise records 221, and payment method records 209 or 213.
- 20 • Search users routine 1303 searches for user profiles by name or by taxonomic categories, i.e., by the kinds of expertise or education the user has.
- view user info 1303 permits a user to view information from a profile table and from the education, expertise, and rating tables available through the profile table.
- rate user info 1305 permits a user to rate another user's performance either as a service
 25 requester or as a service provider with regard to a task. The routine writes the rating to the provider state record for the user, profile, and task and also updates the relevant expertise rating record 229, 233, or 235 for the profile.

Task routines 117

- 30 • post task routine 1307 makes a provider state record 261 and a task record 249 for the user, profile tuple that is posting the task.
- view task routine 1309 displays information from a task record 249 and from pricing scheme record 309 for the task record.

- search task routine 1311 searches for tasks by the taxonomic categories in the task records 249.
- bid on task 1313 creates a provider state record 261 for a provider, profile, task tuple, and makes a negotiation state record 313 for the tuple, and sends a message to the requester, profile, task tuple. In sending the message, send message routine 1325 is used and a message record 409 is created.
- respond to bids 1315 alters pricing scheme record 309 to reflect the current state of the bidding, and when a bid is accepted, uses send message routine 1325 to send a message to the successful provider, profile tuple.
- pay provider 1317 uses the information in pricing scheme record 309 for the task, in provider payment method table 211 for the provider, profile tuple, and in activity record 207 for the requester, profile, task tuple to pay the provider for the service.

Solution routines 118

- post solution routine 1319 creates a solution record 269, solution history record 273, and if necessary, a solution file record 317 for the solution for a provider, profile, task tuple and uses send message 1325 to send a message to the task's requester, profile tuple to indicate that a solution has been provided.
- view solution routine 1321 views a solution for a task represented by a solution record 269, which may involve viewing the file specified by solution file record 317. In some instances, the solution will be ambiguous.
- accept/reject solution routine 1323 creates a solution history record 273 and if the solution is accepted, uses pay provider routine 1317 to pay the successful provider, profile tuple. With either acceptance or rejection, the routine uses send message 1325 to send a message to the solution's provider, profile tuple.

Message routines 119

- send message 1325 sends a message to a user with regard to a task, creating a message record 409 as it does so.
- view messages 1327 permits a user, profile, task tuple to view messages relevant to it and the task.

Archive routines 120

- search archives 1329 uses search users 1302, search task 1311, view user info 1303, view solution 1321, and view message 1329 to view archived information in database 125.

Details of the user interface for system 101

- 5 As previously explained, the user interface for system 101 is Web pages. In the following, exemplary portions of the user interface will be explained and related to database 125.

The initial screen: FIG. 14

- 10 When a user inputs the server address of server 109 into his or her browser, the first Web page he or she receives from server 109 is page 1401 shown in FIG. 1. This page explains what the service dispatch system is, has a portion 1403 which permits already-registered users to log in and others to register, and has a tab bar 1405 for tabs for initial Web pages of other portions of the user interface for the service dispatch system.

Registration: FIGs. 15 and 16

- 15 Figs. 15 and 16 show the registration interface. Fig. 15 shows interface 1501 which collects a username and a password at 1503, an email address for use by system 101 at 1505, and a specification at 1507 concerning how emails are to be received. The information specified here becomes part of account record 205 for the new user. FIG. 16 shows interface 1601
20 which collects the actual name at 1603 and location information at 1607 of the user and requests the name for a default profile record 225 at 1605. As shown in interface 1601, system 101 protects the privacy of its users by never revealing the actual name of the user contained in account record. As shown at 1607, the user can select the parts of his or her location information which are to be publicly available.

25

Posting a task: FIGs. 17 and 18

- FIGs. 17 and 18 show the interface that a service requester uses for posting a task. FIG. 18 shows interface 1801 for defining the task. The task here is a question. At 1803, the task requester inputs the task title, a category for the task, a posted date, a price, and a description of
30 the task. As seen at 1803, the task requester can also specify a length of time left to respond to the task and can clarify the task in response to queries by service providers.

FIG. 17 shows the interface 1701 used to categorize the task and specify the price. At 1703, the service requester provides a primary category 1704 and selects from a list of subcategories 1705. At 1706, the requester further characterizes the programming environment that the question is about. This information goes into task record 249 for the task. At 1707, the requester has set a fixed price of \$30.00 for the task and system 101 is setting forth the payment terms. The information for the price is stored in pricing scheme record 309 belonging to the task.

At 1807 and 1813 in FIG. 18 are shown the interfaces by means of which a service requester keeps track of solutions of the task. Bar 1807 indicates that the task's status is solved; the buttons at 1811 permit the service requester to add a clarification if the solutions show that one is needed and to pay the service provider(s) if the results are satisfactory. At 1813 is shown the list of solutions which service providers have provided for the question and messages concerning the question. The information here comes from solution records 269 and message records 409 for the task.

Assigning a task: FIGs. 35-39

When either the requester or system 101 has selected one or more profiles of service providers to solve the task, system 101 makes a provider state record 261 for each of the selected profiles so that the task is related to the profile and then sends the provider a message indicating that he or she has been selected. When the provider, in this case the one with the profile name "hoops", looks at the tasks for the profile, task notification interface 3501, shown in FIG. 35, appears. At 3503, the interface describes the task, the posting profile, the time of posting, the time left to perform the task, and the price. The information is of course from the requester's profile, the provider state record 261 for the requester, profile, task tuple, and the task record 249 for the task. Since the provider has as yet taken no action relative to the task, status field 3505 indicates that the status of the task for the provider is *uninvolved*. The buttons at 3505 let the provider either request to be assigned to the task or permits him or her to send a message to the requester with questions. At 3507 is shown the interface where the provider receives messages and writes his or her solution to the task.

When the provider clicks on the *request assignment* button, interface 3601, shown in FIG. 36 appears. There, at offer amount 3603, the provider inputs how much he or she wants to do the

task; at 3605, he or she writes a note indicating what he or she proposes to do. When he or she is satisfied, he or she clicks on the "submit" button. As a result of this action, system 101 makes a message record 409 for the reply, updates negotiation state record 313 for the provider/profile/task tuple, and makes a status history record 253 indicating the transaction for the provider/profile/task tuple.

FIG. 37 shows the interface 3701 that the requester uses to act on the provider's request to have the task assigned to him or her. At 3703, there is a table of action items for the task's requester profile. At 3705, there appears an item that indicates that the profile "hoops" has placed a bid of \$33.00 to do the prime number computation task. At 3707, there is a table listing new messages received by the requester profile; shown at 3709 is the text of the message that "hoops" sent when he or she requested assignment. At 3711 appears a table of the status of recent tasks. The entry for the prime number task appears at 3713. The status of the prime number task for the service requester profile is "pending", since no bids have as yet been accepted.

FIG. 38 shows the interface that appears when the service requester clicks on the underlined portion of row 3705. Interface 3801 provides the task description including the current price at 3803; at 3805 is indicated that the task has the status "pending"; the *add clarification* button permits the service requester to add a clarification to a pending task. Under *solutions and messages* is seen the result of "hoops" assignment request: there appear his or her profile name, the requested price, and the message 3811 that accompanied the assignment request. All of this information comes of course from the records made when the provider requested that the task be assigned to him or her. The service requester uses the buttons at 3809 to accept or reject the offer. Here, the requester is going to accept, so he or she clicks on the *thumbs up* button. The result of this is an updating of the requester, profile, task tuple's provider state record 261 and negotiation state record 313 to indicate the new status of the task and the creation of a new status history record 253 for the requester, profile, task tuple. System 101 then notifies the provider profile of the assignment.

FIG. 39, finally shows the interface 3901 that the provider uses to accept the assignment. To do so, the provider simply clicks on submit button 3907; if he or she desires to send a message along with the acceptance, he or she writes the message in box 3905. Clicking on submit

button 3907 changes the status of the task to "assigned" in provider state record 261 for both the requester, profile, task tuple and the provider, profile, task tuple. Status history records 253 are also created for the two provider state records 261 and for task record 249.

5 Accepting or rejecting a solution: FIG. 19

Interface 1901 is the interface for accepting or rejecting a solution. This interface appears when the service requester clicks on one of the solutions listed at 1813 in FIG. 18. At 1903 is shown the description of the solution, including the name of the service provider profile which submitted the solution, the time of submittal, and the text of the solution. For longer solutions, 10 file uploads 1905 indicates where the service requester can find the solution. This information is from solution record 269 for the solution, and in the case of the file names, from solution file records 317. At 1907, the service requester may send a reply message to the service provider profile regarding the solution. At 1909, the service requester can respond to the solution by requesting a revision, sending a message, or simply rejecting the solution. In any case, the 15 interaction in this screen results in the creation of a status history record 253 for the task. Acceptance of solutions for a task occurs when the service requester selects the pay button at 1811 in FIG. 18. At that point, all solutions which have not been rejected are accepted.

Rating the service providers: FIG. 20

20 Interface 2001 is the interface for rating the service providers who provided solutions for the task. As seen at 2003, each provider has a row in a table, identified by the profile name for the profile the provider used in solving the task. The columns in the table permit the requesting user to rate the quality and timeliness of the solution and to provide a comment. This information goes into provider state record 261 for the profile and task and is periodically 25 summarized for the profile in records in rating tables 229 and 233.

Allocating payments among service providers: FIG. 21

A service requester may specify a maximum number of solutions for a fixed-price task and may divide the fixed price among the service providers. Additionally, the service provider 30 may add a tip to the payment amount. Interface 2101 is the allocation and tip interface. Table 2103 has a row 2105 for each of the service providers who solved the task. Each row has a set of buttons for assigning relative payments and a specification of how much the service provider will receive given the current allocation. Each row also has a field for specifying a tip

for the responder and a field for the total payment owed. Once the service requester has a satisfactory allocation, he or she enters the allocation and tips by pressing button 2107. The result of so doing is to set values in provider state record 261 for each of the user, profile tuples which provided a solution.

5

Sending messages: FIG. 22

FIG. 22 shows the messaging interface 2201 for sending messages between users of system 101. Here, as shown at 2203, the service provider with the profile named *ganesh* has sent a message regarding the task *question to ganesh* to the task's service requester. The service requester is writing a reply in box 2205; when finished, he or she can click on button 2207 to send the message to the service requester. The message from *ganesh* is of course contained in a message record 409, and the reply creates a new message record 409.

10

Posting a project: FIGS. 23-25

Here, the project is a Java applet. Interface 2301 describes the new project. At 2303 the service requester specifies the name of the profile that is posting the project; at 2305, the service requester inputs the project title is input; at 2306, he or she inputs project details. At 2309, the service requester selects a broad category for the project.

15

Continuing with FIG. 24, there is shown the interface for selecting the manner in which the project will be priced. As shown at 2401, a preferred embodiment permits the service requester to specify that the project has a fixed, non-negotiable price, a fixed, negotiable price, or will be set by one of three bidding methods: reverse auction, sealed bidding, and Dutch auction. Here, the service requester has selected sealed bidding.

20

25

In FIG. 25 are shown interface 2501 for selecting subcategories to which the project belongs and interface 2509 for further specifying the price. In interface 2503, the broad category selected at 2309 is displayed, and list 2505 shows subcategories of the category displayed at 2509. The service requester then characterizes the project by selecting one or more of the subcategories and specifying a programming environment for the code at 2507. At 2509, the service requester inputs the maximum price at 2511 and the duration of bidding at 2513.

30

In response to the above inputs of information by the service requester, system 101 creates a new task record 249 for the task, a provider state record 261 for the service requester's profile and the task, a task category record 305 for the task, and a pricing scheme record 307 for the task.

5

Task status interface for the service requester: FIGs. 26 and 34

FIG. 26 shows task status interface 2601 for the service requester. The buttons at 2603 and 2605 permit the user to post new projects or questions using the interfaces explained above. Table 2607 contains a row for each open task posted by the service provider. The row
10 specifies the date the task was posted, the time left to completion, the task status, the task name, the category to which it belongs, and the price. The *Released* status here indicates that the task has been posted, but no one has yet bid on it. The information comes of course from the task record 249 and pricing scheme record 309 for the task. At 2609 is displayed a list of any closed tasks.

15

FIG. 34 shows the interface 3401 that the service requester uses to view messages for the tasks that are relevant to the service provider. Table 3403 has a row for each open task message, with columns for an action taken by the recipient, the date of the message, the source and recipient, the message, and the name of the task it belongs to. As can be seen from the source-
20 recipient column, messages may be exchanged between the service requester and the service providers or between the system and the service requester or service providers. The information in the table comes from the message records 409 for the tasks.

Service provider selection interface: FIG. 27

25 System 101 will itself select one or more service providers for a service requester, with the selection being based on the categories assigned to the task by the service requester and the rating information maintained by system 101 about service providers, but also has the interface 2701 shown in FIG. 27 for permitting the service requester to select the service provider. At 2703, the service requester may select the service provider by the name of a profile belonging
30 to the service provider; otherwise, the user may select categories from list 2705 indicating the kinds of expertise desired in the service provider, and system 101 will provide a list of service providers who have at least one of the kinds of expertise. A portion of the returned list is shown at 2709; each service provider has a row on the list and the row specifies the provider's

profile name, his or her expertises, his or her time with system 101, and his or her current rating. System 101 selects the service providers on the basis of taxonomic categories field 809 in their profile record 225 and the information in table 2709 comes from profile record 225 for the service provider and from the associated rating records 229 and 223 and expertise records 221.

Service provider-task interface: FIGs. 28-30

FIGs. 28 and 29 show the interface that a service provider uses to view the tasks that are available to a specific one of his or her profiles. In FIG. 28, interface 2801 permits the service provider to see the tasks that have been offered specifically to the profile (2803), tasks that are available to be done by this profile (2805), and tasks that the profile is involved with (2807). *Involved* here means that the profile has done some action with regard to the task. This information comes from the provider state records 261 for the profile and the task record 249 associated with each provider state record. At 2809, the service provider can view the tasks completed by the profile. Fig. 29 shows interface 2901 that the service provider can use to view the messages concerning a task that have been sent or received by the profile. The service provider can view open task messages at 2903 and closed task messages at 2905. The information comes from the message records 409 for the tasks with which the profile is involved.

FIG. 30 is the interface that a service provider uses to browse tasks generally. Interface 3001 includes a search interface that permits the service provider to search for a task by pressing button 3007. The search may be by task ID, as shown at 3005, or by category, as shown at 3009. In addition, interface 3001 includes a list 3011 of recently posted tasks which are available for solution by any service provider. Each task has a row in the list, with the row for a task including the posting date, the task's title, and the price. The information in the table row comes from the task record 249 for the task and the pricing scheme record for the task.

Editing account records and profile records: FIGs. 31 and 32

FIG. 31 shows the interface 3101 that a user of system 101 uses to edit his or her account record 205 and records related thereto. At 3103, the user gains entry to the interface for changing the login and contact information in account record 205; at 3105, the user can edit buyer payment method 209 for the account record; at 3107, the user can edit provider payment

method record 213 for the account record; at 3107, the user can also edit the education records 271 and the expertise records 221 associated with the account record.

FIG. 32 shows the interface 3201 that the user can employ to edit a profile record 225
5 belonging to the user's account record 205. As shown at 3203, the user may make a new profile, edit an existing one, or make a default profile. At 3204 is shown a list of profiles belonging to the user. For each profile is listed the profile's name, the associated expertises, the date the profile was made, and the ratings for the profile as provider and buyer. The information being edited is contained in profile record 225 and the associated rating records
10 229, 233, and 237.

The interface for viewing the user's monetary account with system 101: FIG. 33

FIG. 33 shown interface 3101 for viewing the user's monetary account with system 101. At 3303 is shown the table which summarizes the monetary transactions in the account, and at
15 3305 is the interface for viewing a list of activities for the current account. The information in this table comes from the activity records 257 for a user.

Conclusion

The inventors of the present invention have disclosed to those skilled in the arts with which the
20 service dispatch system is concerned the best method presently known to them of implementing the system. However, as will be immediately apparent to those skilled in the relevant arts, there are many ways of implementing the principles of the service dispatch system other than the one disclosed herein, and the implementation may vary according to the kinds of services being requested and provided. For example, in some implementations,
25 service requesters and providers may interact directly with each other, rather than by means of profiles. In other implementations, the tasks may all have fixed prices, and in still others, bargaining techniques may be used other than those disclosed in the preferred embodiment. Where the system is specialized to a particular small set of tasks, there may be no need for task categorization and matching of service requester and service provider by category and
30 expertise. Other embodiments may also use other techniques for selecting a set of service providers to whom the task may be assigned. For example, the system might maintain a list of currently-available service providers that is ordered by date of last task completion and simply select the service provider who had gone longest without an assignment.

The techniques used for ambiguating task solutions and for dealing with unsatisfactory solutions may also vary with the kind of system. For example, with simple tasks, the service requester may be permitted only to accept or reject a solution. Payment systems can include
5 any mechanism for ensuring that an accepted solution is paid for, and an embodiment may employ any mechanism for preserving information which provides the necessary audit trails for the embodiment. For example, where the solution is simple and there is no price negotiation, audit trails may be needed only to determine how much the service requester owes the system and how much the system owes the individual service providers.

10

While the Internet provides a particularly advantageous environment for users of the service dispatch system, the system can be used in any environment that gives the system an interactive interface to the user

15 For all of the foregoing reasons, the *Detailed Description* is to be regarded as being in all respects exemplary and not restrictive, and the breadth of the invention disclosed herein is to be determined not from the *Detailed Description*, but rather from the claims as interpreted with the full breadth permitted by the patent laws.

20 **What is claimed is:**

1 1. A method wherein a service requester employs a service dispatch system to obtain a
2 solution to a task for a price, the requester having access to the service dispatch system via an
3 interactive interface and the method comprising the steps performed using the interactive
4 interface of:

5 sending the service dispatch system a description of the task;
6 receiving an indication from the service dispatch system that the solution is available
7 from at least one of a set of potential service providers assigned to the task by the service
8 dispatch system; and
9 indicating to the service dispatch system that the price is to be paid to the service
10 providers in the set from whom the solution is available.

1 2. A method wherein a service provider employs a service dispatch system to provide a
2 service requestor with a solution to a task for a price, the provider having access to the service
3 dispatch system via an interactive interface and the method comprising the steps performed
4 using the interactive interface of:

5 receiving a notification from the service dispatch system that the task is available for
6 solution;
7 indicating to the service dispatch system that the solution is available; and
8 receiving the price from the service dispatch system.

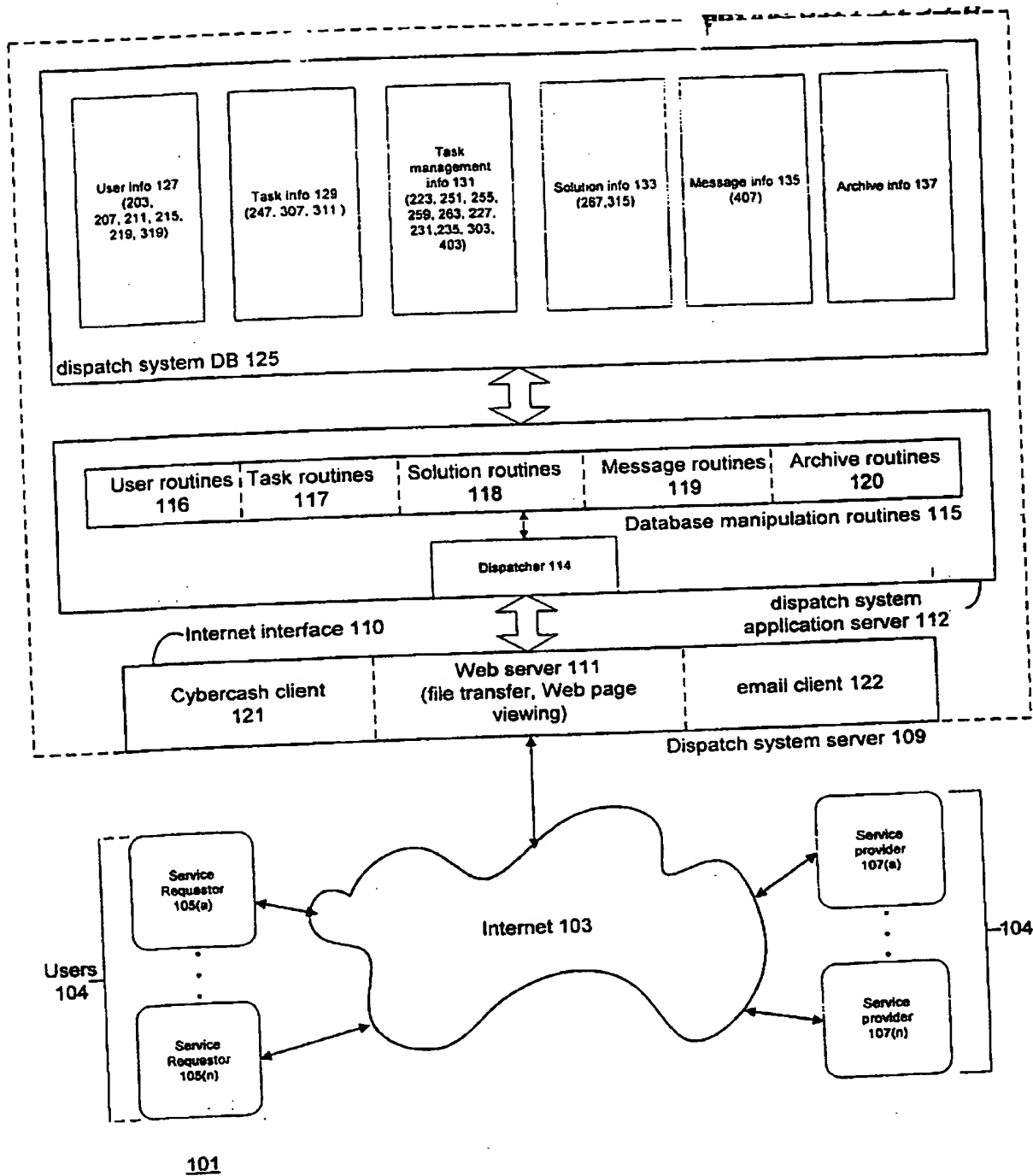


FIG. 1

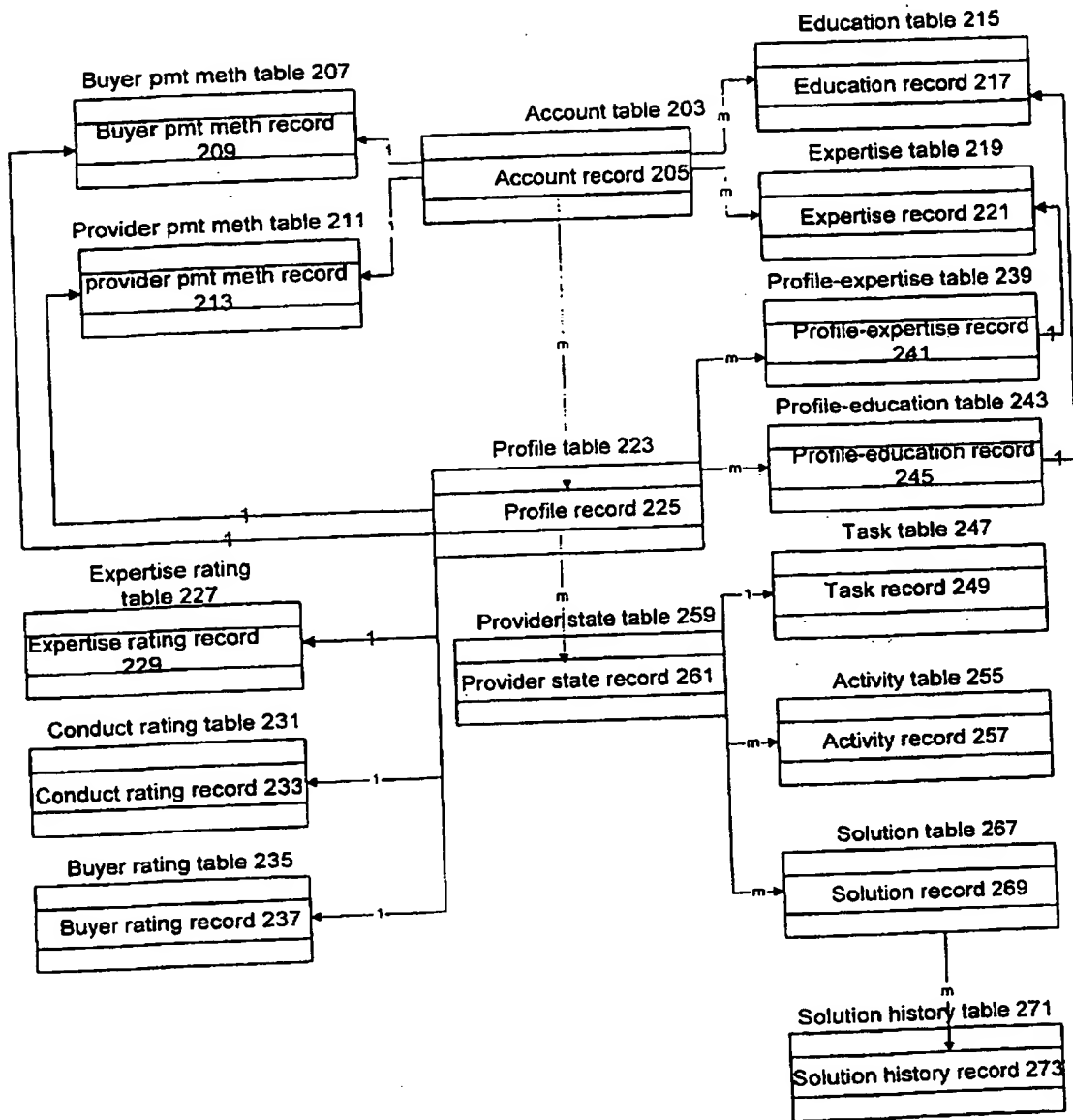
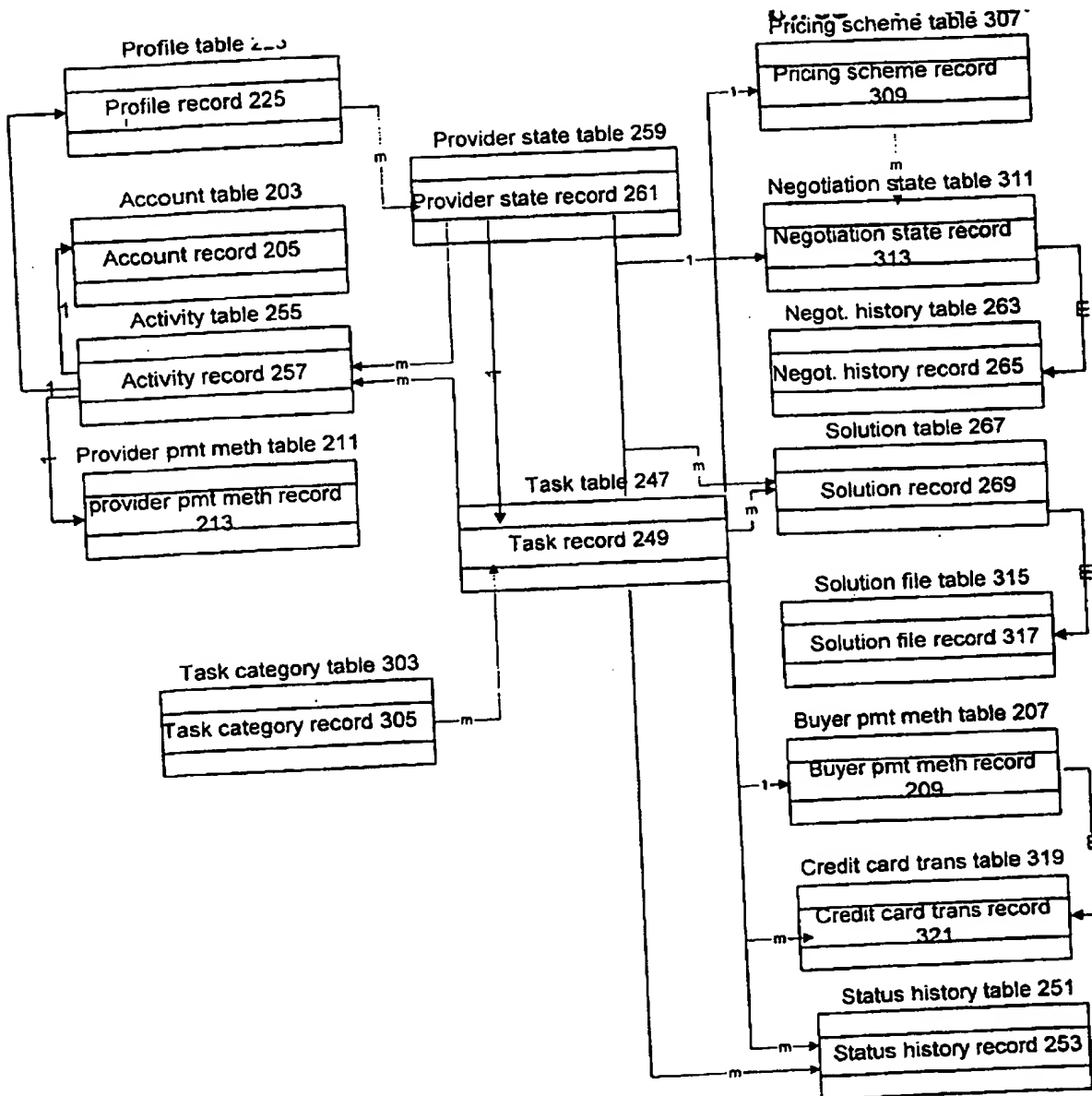


FIG. 2 201



301

FIG. 3

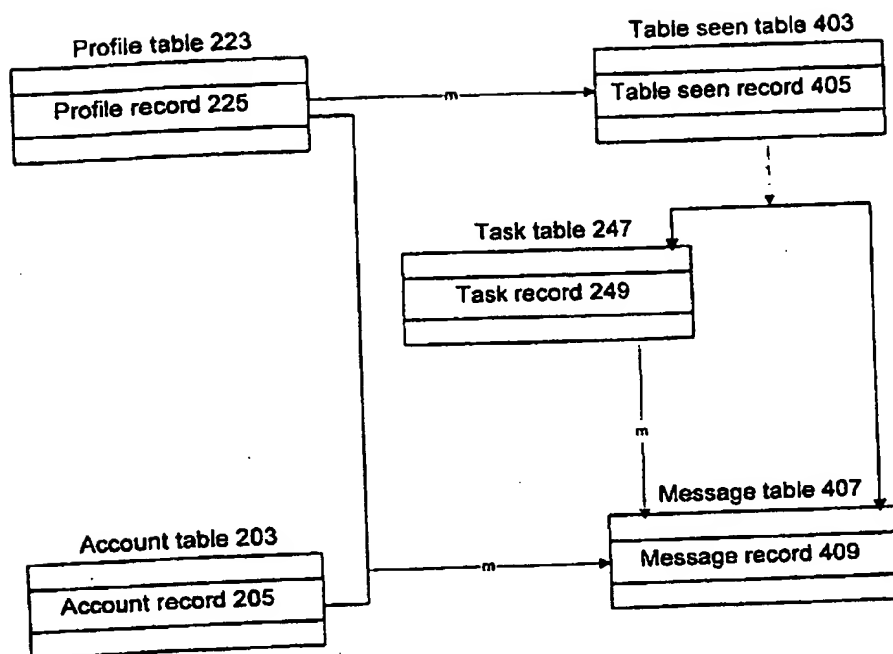
401

FIG. 4

```

CREATE TABLE tbl_hd_account (
  hd_account_id      INTEGER NOT NULL,
  mod_count          INTEGER NOT NULL,
  time_entered       CHAR(12) NOT NULL,
  time_updated       CHAR(12) NOT NULL,
  username            CHAR(20) NOT NULL,
  password            CHAR(20) NOT NULL,
  taxonomic_categories CHAR(40) NULL,
  first_name          CHAR(20) NOT NULL,
  middle_initial      CHAR(1) NULL,
  last_name           CHAR(20) NOT NULL,
  profile_name        CHAR(20) NULL,
  street              CHAR(40) NULL,
  street2             CHAR(40) NULL,
  city                CHAR(20) NULL,
  state               CHAR(20) NULL,
  zip_code            CHAR(10) NULL,
  country             CHAR(20) NULL,
  email               VARCHAR(100) NULL,
  email2              VARCHAR(100) NULL,
  receive_email       CHAR(1) NULL,
  buyer_pay_method_id INTEGER NULL,
  provider_pay_method_id INTEGER NULL,
  birth_year          CHAR(4) NULL,
  nationality          CHAR(20) NULL,
  gender              CHAR(1) NULL,
  visible             CHAR(20) NULL,
  status              CHAR(1) NULL,
  current_profile_id  INTEGER NULL,
  referer             CHAR(10) NULL,
  question            VARCHAR(100) NULL,
  answer              VARCHAR(100) NULL,
  referrer_id         CHAR(10) NULL,
  referral_id         CHAR(10) NULL,
  balance             INTEGER NULL,
  expertises_time     integer null,
  educations_time     integer null,
  escrow_total        INTEGER NULL,
  escrow_available    INTEGER NULL,
  profiles_time       integer null,
  tasks_time          integer null,
  status_history_time integer null,
  activities_time     integer null,
  upload_directory    CHAR(32) null
);

```

Diagram illustrating the structure of the `tbl_hd_account` table, with fields grouped into categories indicated by brackets and numbered boxes:

- 501: `hd_account_id`
- 503: `mod_count`
- 505: `time_entered`, `time_updated`, `username`, `password`, `taxonomic_categories`, `first_name`, `middle_initial`, `last_name`, `profile_name`, `street`, `street2`, `city`, `state`, `zip_code`, `country`, `email`, `email2`, `receive_email`
- 507: `buyer_pay_method_id`, `provider_pay_method_id`
- 509: `birth_year`
- 510: `nationality`
- 511: `gender`
- 513: `visible`, `status`
- 515: `current_profile_id`
- 517: `referrer`, `question`
- 518: `answer`
- 517: `referrer_id`, `referral_id`
- 519: `balance`, `expertises_time`, `educations_time`, `escrow_total`, `escrow_available`, `profiles_time`, `tasks_time`, `status_history_time`, `activities_time`, `upload_directory`

205

FIG. 5

WO 00/29989

6 / 40

```

CREATE TABLE tbl_education (
  education_id      INTEGER NOT NULL,
  mod_count         INTEGER NOT NULL,
  time_entered      CHAR(12) NOT NULL,
  time_updated      CHAR(12) NOT NULL,
  owner_id          INTEGER NOT NULL,
  institution        CHAR(40) NULL,
  starting_year     CHAR(4) NULL,
  ending_year       CHAR(4) NULL,
  degree_subject    CHAR(20) NULL,
  degree_level      CHAR(24) NULL,
  degree_year       CHAR(4) NULL,
  visible           CHAR(1) NULL
);

```

609
611
613
615

217

```

CREATE TABLE tbl_expertise (
  expertise_id      INTEGER NOT NULL,
  mod_count         INTEGER NOT NULL,
  time_entered      CHAR(12) NOT NULL,
  time_updated      CHAR(12) NOT NULL,
  owner_id          INTEGER NOT NULL,
  primary_categ     CHAR(10) NOT NULL,
  secondary_categs  CHAR(50) NULL,
  expertise_notes    VARCHAR(4000) NULL,
  expertise_duration CHAR(20) NULL,
  certifications    VARCHAR(1000) NULL,
  visible           CHAR(1) NULL
);

```

601
603
605
607

221

FIG. 6

```

CREATE TABLE tbl_pay_method (
  pay_method_id      INTEGER NOT NULL,
  mod_count          INTEGER NOT NULL,
  time_entered       CHAR(12) NULL,
  time_updated       CHAR(12) NULL,
  class              CHAR(1) NULL,
  credit_card_company CHAR(20) NULL,
  credit_card_number CHAR(16) NULL,
  credit_card_exp_date CHAR(8) NULL,
  credit_card_holder  VARCHAR(100) NULL,
  credit_card_zip_code CHAR(10) NULL,
  check_street        CHAR(40) NULL,
  check_street2       CHAR(40) NULL,
  check_city          CHAR(20) NULL,
  check_state         CHAR(20) NULL,
  check_zip_code      CHAR(10) NULL,
  check_country       CHAR(20) NULL,
  bank_name           CHAR(40) NULL,
  account_number      CHAR(20) NULL,
  swift_aba_number    CHAR(20) NULL,
  auth_price          INTEGER NULL
);

```

Diagram illustrating line numbers 701, 703, 705, and 707 grouped by brackets next to the SQL code.

209, 213

FIG. 7

WO 00/29989

8 / 40

```

CREATE TABLE tbl_profile (
  profile_id          INTEGER NOT NULL,
  mod_count           INTEGER NOT NULL,
  time_entered        CHAR(12) NOT NULL,
  time_updated        CHAR(12) NOT NULL,
  owner_id            INTEGER NOT NULL,
  name                CHAR(40) NOT NULL,
  taxonomic_categories CHAR(40) NULL,
  email               VARCHAR(100) NULL,
  email2              VARCHAR(100) NULL,
  inherit_emails      CHAR(1) NULL,
  buyer_pay_method_id INTEGER NULL,
  provider_pay_method_id INTEGER NULL,
  visible             CHAR(20) NULL,
  description          VARCHAR(4000) NULL,
  expertise_rating_id  INTEGER,
  conduct_rating_id    INTEGER,
  buyer_rating_id     INTEGER,
  participation_rating CHAR(10) NULL,
  buyer_participation_rating CHAR(10) NULL,
  earliest_task_time   CHAR(12) NULL,
  latest_task_time     CHAR(12) NULL,
  expertises_time      integer null,
  educations_time      integer null,
  tasks_time           integer null,
  status_history_time  integer null,
  buyer_messages_time  integer null,
  provider_messages_time integer null,
  provider_states_time integer null
);

```

225

```

CREATE TABLE tbl_rating(
  table_rating_id    INTEGER NOT NULL,
  mod_count          INTEGER NOT NULL,
  time_entered        CHAR(12) NOT NULL,
  time_updated        CHAR(12) NOT NULL,
  class              CHAR(1) NULL,
  rating_total        INTEGER NULL,
  rating_count        INTEGER NULL,
  rating_average      NUMBER NULL
);

```

229, 233, 237

FIG. 8

```

CREATE TABLE tbl_task (
  task_id
  mod_count
  time_entered
  time_updated
  owner_id
  structured_category
  dispatch_prototype
  name
  description
  url
  public_directory
  pricingscheme_id
  buyer_pay_method_id
  buyer_status
  record_state
  class
  number_of_bindings
  seen_time
  max_incidents
  associated_contract_id
  provider_id
  commission_key
  provider_states_time
  status_history_time
  solutions_time
  transaction_id
  clarification_sp
  direct_notifies
  solutions_count
);

```

```

  INTEGER NOT NULL,
  INTEGER NOT NULL,
  CHAR(12) NOT NULL,
  CHAR(12) NOT NULL,
  INTEGER NOT NULL,
  CHAR(40) NULL,
  CHAR(10) NULL,
  CHAR(60) NOT NULL,
  VARCHAR(4000) NOT NULL,
  CHAR(40) NULL,
  CHAR(1) NULL,
  INTEGER NOT NULL,
  INTEGER NULL,
  CHAR(1) NULL,
  INTEGER NULL,
  CHAR(1) NOT NULL,
  INTEGER NULL,
  CHAR(12) NULL,
  INTEGER NULL,
  INTEGER NULL,
  INTEGER NULL,
  CHAR(20) NULL,
  integer null,
  integer null,
  integer null,
  INTEGER NULL,
  INTEGER NULL,
  VARCHAR(400) null,
  INTEGER NULL

```

249

```

CREATE TABLE tbl_task_solution (
  task_solution_id
  mod_count
  time_entered
  time_updated
  task_id
  owner_id
  status
  aux_status
  solution_text
  files
  seen_time
  status_history_time
  solution_history_time
);

```

```

  INTEGER NOT NULL,
  INTEGER NOT NULL,
  CHAR(12) NULL,
  CHAR(12) NULL,
  INTEGER NOT NULL,
  INTEGER NOT NULL,
  CHAR(1) NULL,
  CHAR(1) NULL,
  VARCHAR(4000) NOT NULL,
  CHAR(1) NULL,
  CHAR(12) NULL,
  integer null,
  integer null

```

269

FIG. 9

10 / 40

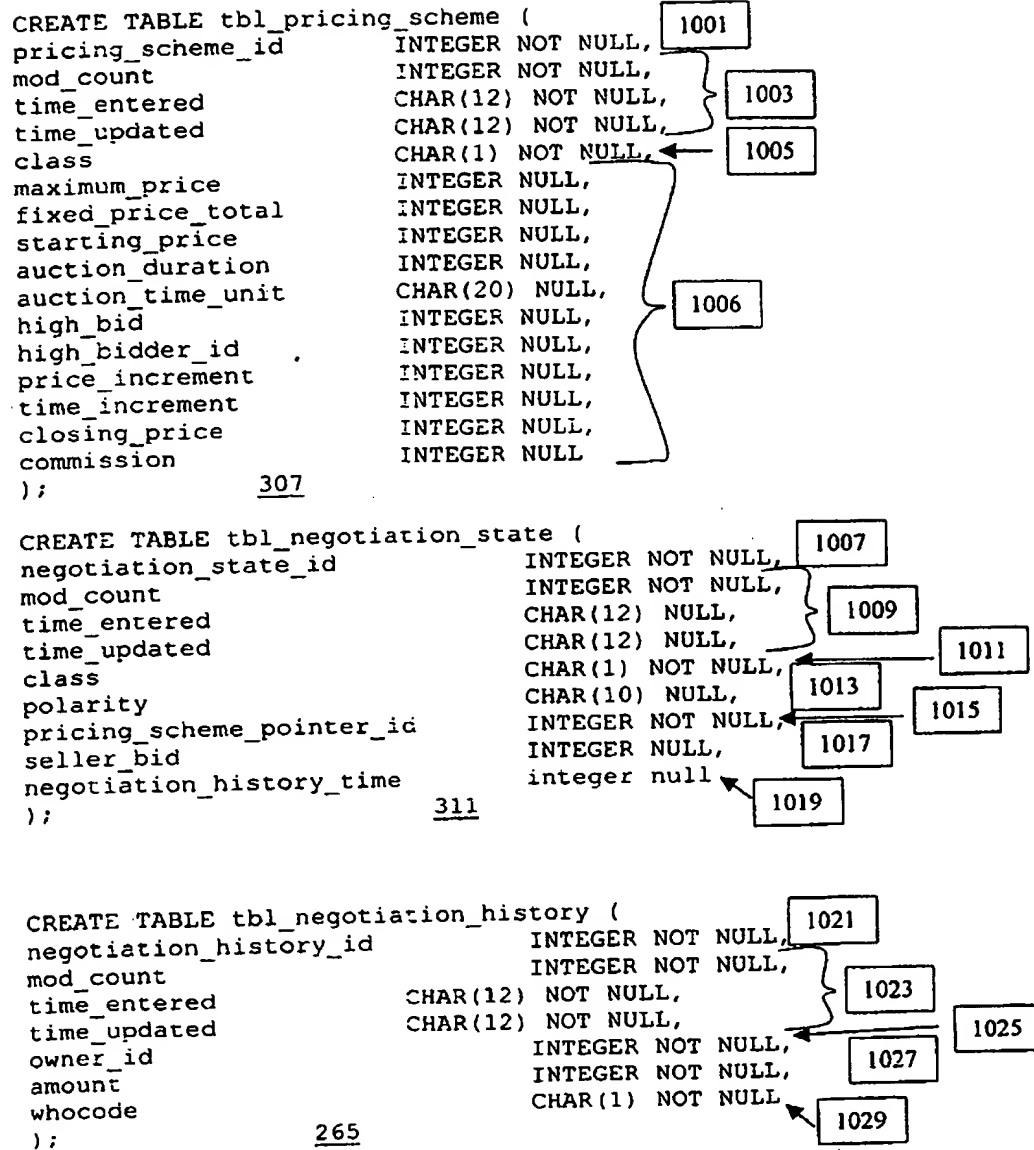


FIG. 10

11 / 40

```

CREATE TABLE tbl_provider_state (
  provider_state_id      INTEGER NOT NULL, 1101
  mod_count              INTEGER NOT NULL,
  time_entered           CHAR(12) NOT NULL, 1103
  time_updated           CHAR(12) NOT NULL,
  task_id               INTEGER NOT NULL, 1105
  owner_id              INTEGER NOT NULL,
  status                CHAR(1) NULL, 1107
  negotiations_id       INTEGER NULL, 1109
  solution_id           INTEGER NULL,
  price_paid            VARCHAR(4000) NULL, 1111
  rating_comment        INTEGER NULL,
  expertise_rating      INTEGER NULL, 1113
  conduct_rating        INTEGER NULL,
  buyer_rating         VARCHAR(4000) NULL, 1115
  buyer_comment        CHAR(10) NULL,
  offer                CHAR(12) NULL, 1117
  seen_time            integer null,
  status_history_time   integer null,
  solutions_time       integer null,
  tip_amount           INTEGER NULL 1119
);

```

261

```

CREATE TABLE tbl_base_activity (
  base_activity_id      INTEGER NOT NULL, 1121
  mod_count            INTEGER NOT NULL,
  time_entered         CHAR(12) NOT NULL, 1123
  time_updated         CHAR(12) NOT NULL,
  owner_id            INTEGER NOT NULL, 1124
  task_id            INTEGER NULL,
  class              CHAR(1) NOT NULL, 1125
  provider_id        INTEGER NOT NULL, 1127
  hd_commission_amount INTEGER NULL,
  other_commission_amount INTEGER NULL, 1129
  other_commission_code CHAR(10) NULL,
  debit_amount       INTEGER NULL,
  credit_amount      INTEGER NULL,
  balance            INTEGER NULL,
  pay_method_id      CHAR(20) NULL,
  type              CHAR(20) NULL,
  label             INTEGER NULL,
  escrow_available   INTEGER NULL,
  escrow_total       INTEGER NULL,
  notes             CHAR(50) NULL
);

```

257

FIG. 11

12 / 40

```

CREATE TABLE tbl_status_history (
  status_history_id  INTEGER NOT NULL,
  mod_count          INTEGER NOT NULL,
  time_entered       CHAR(12) NOT NULL,
  time_updated       CHAR(12) NOT NULL,
  owner_id           INTEGER NOT NULL,
  status             CHAR(1) NULL
);

```

253

```

CREATE TABLE tbl_solution_history (
  solution_history_id  INTEGER NOT NULL,
  mod_count            INTEGER NOT NULL,
  time_entered         CHAR(12) NOT NULL,
  time_updated         CHAR(12) NOT NULL,
  owner_id             INTEGER NOT NULL,
  solution_text        VARCHAR(4000) NULL
);

```

273

```

CREATE TABLE tbl_message (
  message_id          INTEGER NOT NULL,
  mod_count           INTEGER NOT NULL,
  time_entered        CHAR(12) NULL,
  time_updated        CHAR(12) NULL,
  task_id             INTEGER NOT NULL,
  class               CHAR(1) NOT NULL,
  text                VARCHAR(4000) NOT NULL,
  sender_id           INTEGER NOT NULL,
  recipient_id        INTEGER NOT NULL,
  party_regarding_id  INTEGER NULL,
  type                CHAR(1) NULL,
  seen_time           CHAR(12) NULL,
  recipient_delete_code INTEGER NULL,
  sender_delete_code  INTEGER NULL
);

```

409

FIG. 12

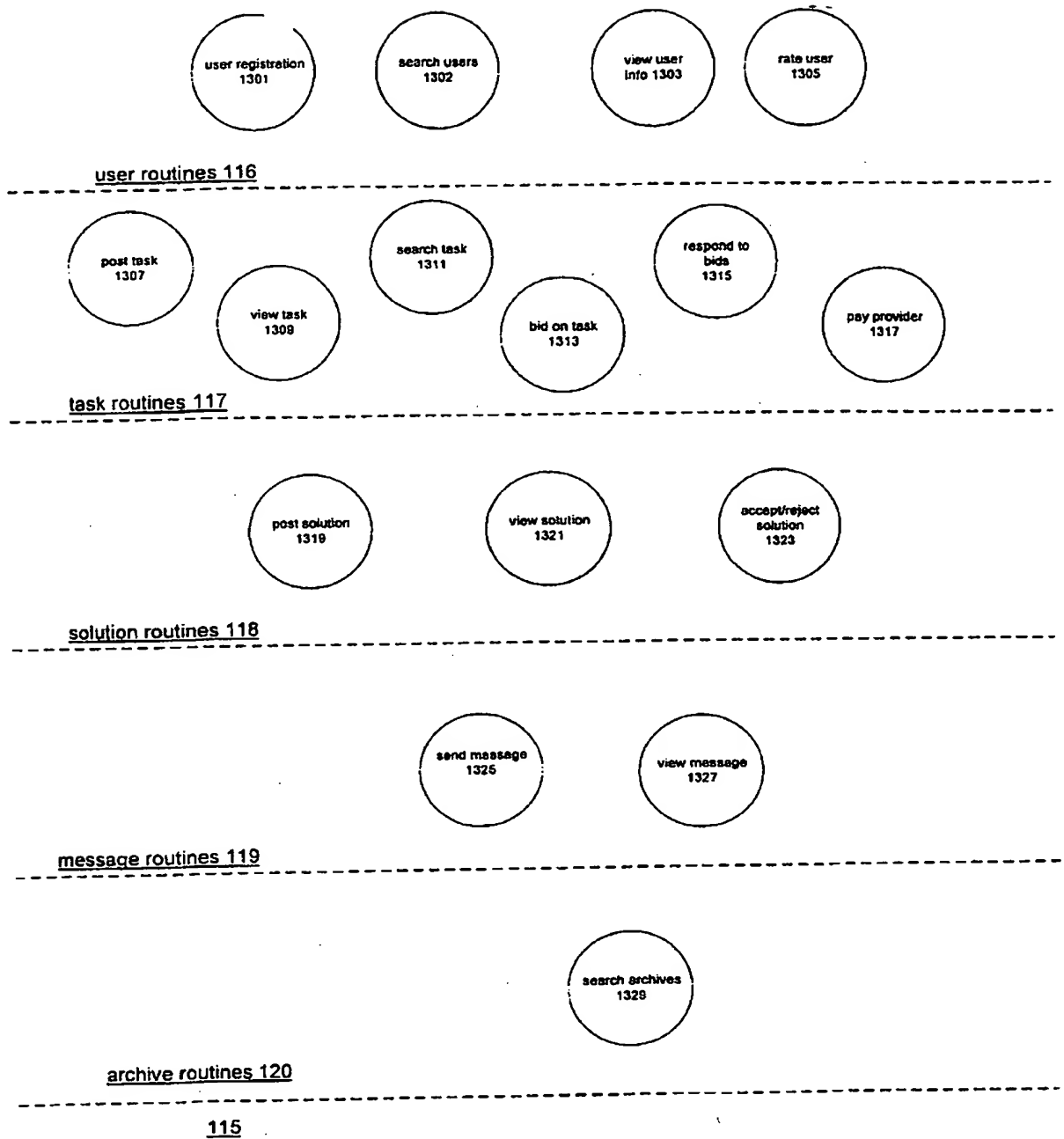


FIG. 13

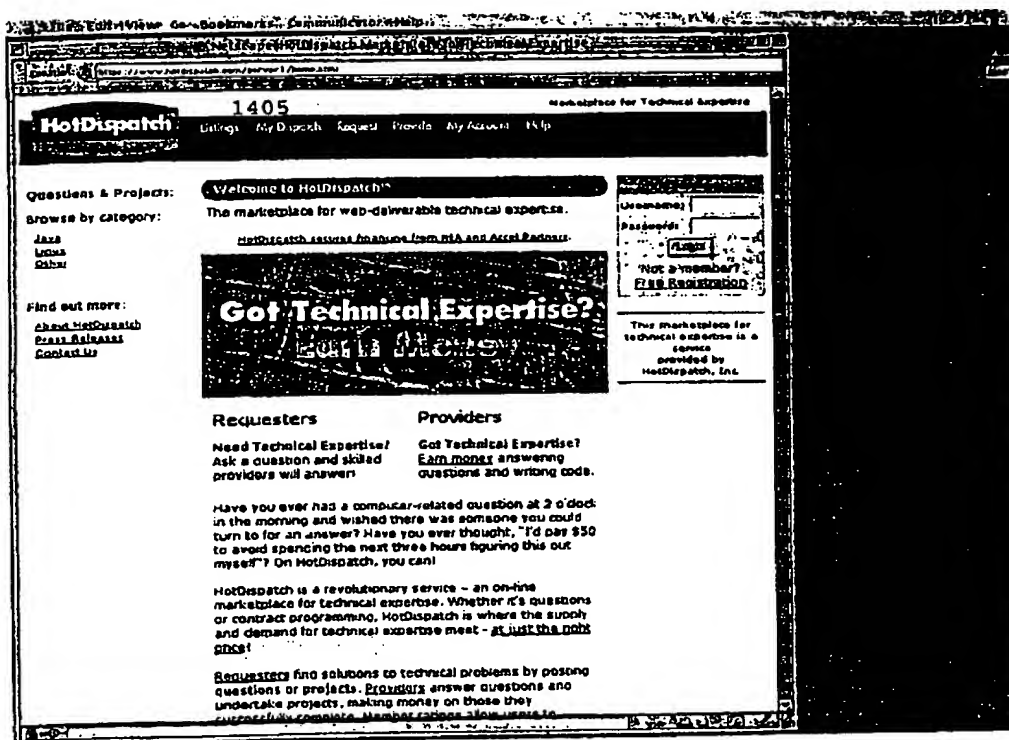


FIG. 14

1401

HotDispatch Home MyDispatch Request Profile My Account Help
[Edit Account](#) [My Profile](#) [My History](#)

REGISTRATION STEP 2 OF 3

Complete the information below to create an account on HotDispatch allowing you to provide and/or request technical expertise. All items with a star (*) are required fields.

Each user establishes one, and only one, login account. The login account enables us to identify you as a HotDispatch user. The information is used internally by HotDispatch and NOT made public (see our [Privacy Statement](#)). However, each user can establish multiple professional profiles. These profiles are your public persona and should contain information useful to other HotDispatch users.

Enter the user name and password that you would like to use for your HotDispatch account

★ User Name: Minimum 5 characters. Your user name is never revealed by HotDispatch to other users or third parties.

★ Password: Minimum 5 characters. Pick a combination of letters and numbers you can remember.

★ Confirm Password: Never share it with another person, even if they claim to be a HotDispatch employee.

★ E-mail Address: 1505 The address you provide here is only used by HotDispatch for notices relating to your account.

Receive e-mails: ☒ Enabled Specify when you would like to receive e-mail messages generated by or via HotDispatch.

☐ Task Notifications

☐ Homebody L. C. ☐ Disability

1501

FIG. 15

HotDispatch! Marketplace for Technical Experts [Log Out](#)

[Logout](#) [My Dispatch](#) [Request](#) [Friends](#) [My Auctions](#) [Help](#)

[Edit Account](#) [My Profiles](#) [My Balances](#)

REGISTRATION STEP 2 OF 5

Who Are You?

The name you enter here is never shown to providers or requesters you do business with on HotDispatch nor is it shared with third parties. We ask for your name to assure uniqueness of accounts (only one HotDispatch account is permitted per person):

1603

★ First Name: This is never made public

Middle Initial: This is never made public

★ Last Name: This is never made public

You communicate with other members on HotDispatch via your "profile" or "screen" name. You can create multiple profiles on HotDispatch, but in the initial registration, you only need to provide a name for your default profile. You can use your real name (e.g. John Smith), adapt it (e.g. JSmith), or create a new one (e.g. bob hope). Names that are offensive are not permitted.

★ Default Profile Name: This is public

1605

Where Are You Located?

Your street address and zip code is never made public or shared with a third party. You can elect to have your city, state, and or country appear as part of your public profile.

1607

Show in Public Profile

☐ Home page: Public only if checked.

☐ ICQ address: Public only if checked.

FIG. 16

1601

17 / 40

File Edit View Go Bookmarks Communicator Help

HotDispatch! Marketplace for Technical Experts Log Out

Logout My Dispatches Register Profile My Account Help

My Tools My Questions Search Questions

1703

Further categorize the question

Primary category:

Software Development / Programming Languages / Java 1704

Subcategories:

<input type="checkbox"/> Algorithms	<input type="checkbox"/> Java Advanced Imaging	<input type="checkbox"/> Mathematics
<input type="checkbox"/> Applets	<input type="checkbox"/> Java Foundation Classes	<input type="checkbox"/> Memory Management/GC
<input type="checkbox"/> AWT	<input type="checkbox"/> Java IDL	<input type="checkbox"/> Multimedia
<input type="checkbox"/> Beans	<input type="checkbox"/> Java Plug-in	<input type="checkbox"/> Networking/Serialization
<input type="checkbox"/> Bugs and Exceptions	<input type="checkbox"/> Java Server Pages	<input type="checkbox"/> Other
<input type="checkbox"/> Collections and Enumerations	<input type="checkbox"/> Java Virtual Machine	<input type="checkbox"/> Profiling & Optimization
<input type="checkbox"/> CORBA	<input type="checkbox"/> Javadoc	<input type="checkbox"/> RMI
<input type="checkbox"/> Dates and Formatters	<input type="checkbox"/> JavaScript	<input type="checkbox"/> Security
<input type="checkbox"/> Design Patterns	<input type="checkbox"/> JDBC	<input type="checkbox"/> Servlets
<input type="checkbox"/> Input/Output	<input type="checkbox"/> JINI	<input type="checkbox"/> Streams & Strings
<input type="checkbox"/> Internationalization	<input type="checkbox"/> JST Compiler	<input type="checkbox"/> Third party libraries
<input type="checkbox"/> Interoperability	<input type="checkbox"/> JNDI	<input type="checkbox"/> Threads
<input type="checkbox"/> JAR Files	<input type="checkbox"/> JMS	<input type="checkbox"/> XML
<input type="checkbox"/> Java 2D	<input type="checkbox"/> Language Basics	
<input type="checkbox"/> Java 3D		

Browser: Netscape Navigator 4.0

IDE: CodeWarrior 4.5

Java Version: 1.1.5

1705

What is the Price?

Fixed price: \$55.00 Enter the price (in dollars) you are willing to pay for the answer to your question. You will receive answers from up to 5 providers. You can allocate the whole amount to the "winning" answer or you can divide it among all or some of the respondents.

1707 Your total cost will be the above amount plus the HotDispatch transaction fee of 15%. If you are paying by credit card, it will be authorized for the full amount, but will only be charged when you close the question.

Confirm your Earlier Information

1701

FIG. 17

1809

HotDispatch: Marketplace for Technical Expenses

Task Title: first question is just a test (task ID: 35967)
 Categories: Other
 Posted: 11/05 4:24 AM
 Price: \$30.00
 Time left: N/A
 Description: this "question" is to just test that direct notification is enabled. "solve" it by posting some acknowledgement in the solution area and you'll receive \$10.

Clarifications: None
 Status: Solved

Solutions and Messages:

Solution Released		1813
11/05 7:07 PM	Henry2 -> HotDispatch	Solution: Give me my money!
11/09 4:36 AM	HotDispatch -> Henry2	Message: Henry2, where did you see this question posted? did you get an e message or otherwise?
11/09 12:44 PM	Henry2 -> HotDispatch	Message: I think I saw it under tasks available to this profile.

FIG. 18

1801

HotDispatch: Marketplace for Technical Expertise

Log Out

My Tasks My Messages Search Providers

Task Solution

Task Title: first question is just a test (task ID: 36330)

To: HotDispatch 1903

Solved by: Henry2

Date: 11/11 10:34 PM

Solution text: Give me my money!

File uploads: 1905

Reply:

1907

Request Revision Send Message Need Solution Click Reply

1909

Current provider: HotDispatch

Last Action: Feedback to HotDispatch

Copyright © 1999 HotDispatch, Inc. Use of this website constitutes acceptance of our Terms and Conditions.

1901

FIG. 19

HotDispatch
Workplace for Technical Experts (Log Out)

My tasks: My Dispatch Report Search Providers

Rate Each Provider

Rate your experience with each provider: 2003

Provider	Excellent	Very Good	Good	Not Good	Comments
Henry2	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	have his stuff
ahh	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	no point above
ahet	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
lasha	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
lanka	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	

Allocate Payments

Allocate the payment to the providers who answered your question. Assign relative payments by using the radio buttons. You may give providers an additional 10% by entering the amount in the top column. (Note that the totals in the right-most column include the 15% HotDispatch fee.)

Provider	100%	50%	25%	10%	0%	Additional	Total
Henry2	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>		

FIG. 20

2001

Allocate Payments

Allocate the payment to the providers who answered your question. Assign relative payments by using the radio buttons. You may give providers an additional fee by entering the amount in the fee column.
(Note that the totals in the right-most column include the 15% HotDispatch fee.)

2103

Provider	Answered	Relative Payment	Additional Fee	Total
2105	<input checked="" type="radio"/>	\$9.00	\$1.00	\$10.00
2106	<input checked="" type="radio"/>	\$9.00	\$1.00	\$10.00
2107	<input checked="" type="radio"/>	\$9.00	\$1.00	\$10.00
2108	<input checked="" type="radio"/>	\$9.00	\$1.00	\$10.00
2109	<input checked="" type="radio"/>	\$9.00	\$1.00	\$10.00
Total		\$45.00	\$5.00	\$50.00

Click here to submit the payments

Current provider HotDispatch: [Local hotline / Feedback to HotDispatch](#)

Copyright © 1999 HotDispatch, Inc. Use of this website constitutes acceptance of HotDispatch's Terms of Service.

FIG. 21

2101

File Edit View Go Bookmarks Communicator Help

[Location: https://www.hotdispatch.com/server1/view-message?D=28577&order=dispatcher&requester=...](#)

[HotDispatch](#)
[Requester Summary](#)
[Provider Summary](#)
[My Dispatch](#)
[Request](#)
[Provide](#)
[My Account](#)
[Help](#)

[Marketplace for Technical Expertise](#)
[log Out](#)

[Refresh](#)

Message

From: ganesh
To: HotDispatch
Date: 11/11 10:37 PM
Regarding task: question to ganesh
Message text: Go ahead and reject this! I dare you!!

2203

Reply to Message

Reply:

Go ahead and reject this! I dare you!!
 how about posting a solution instead of a message?

Send Message

Reset

2205

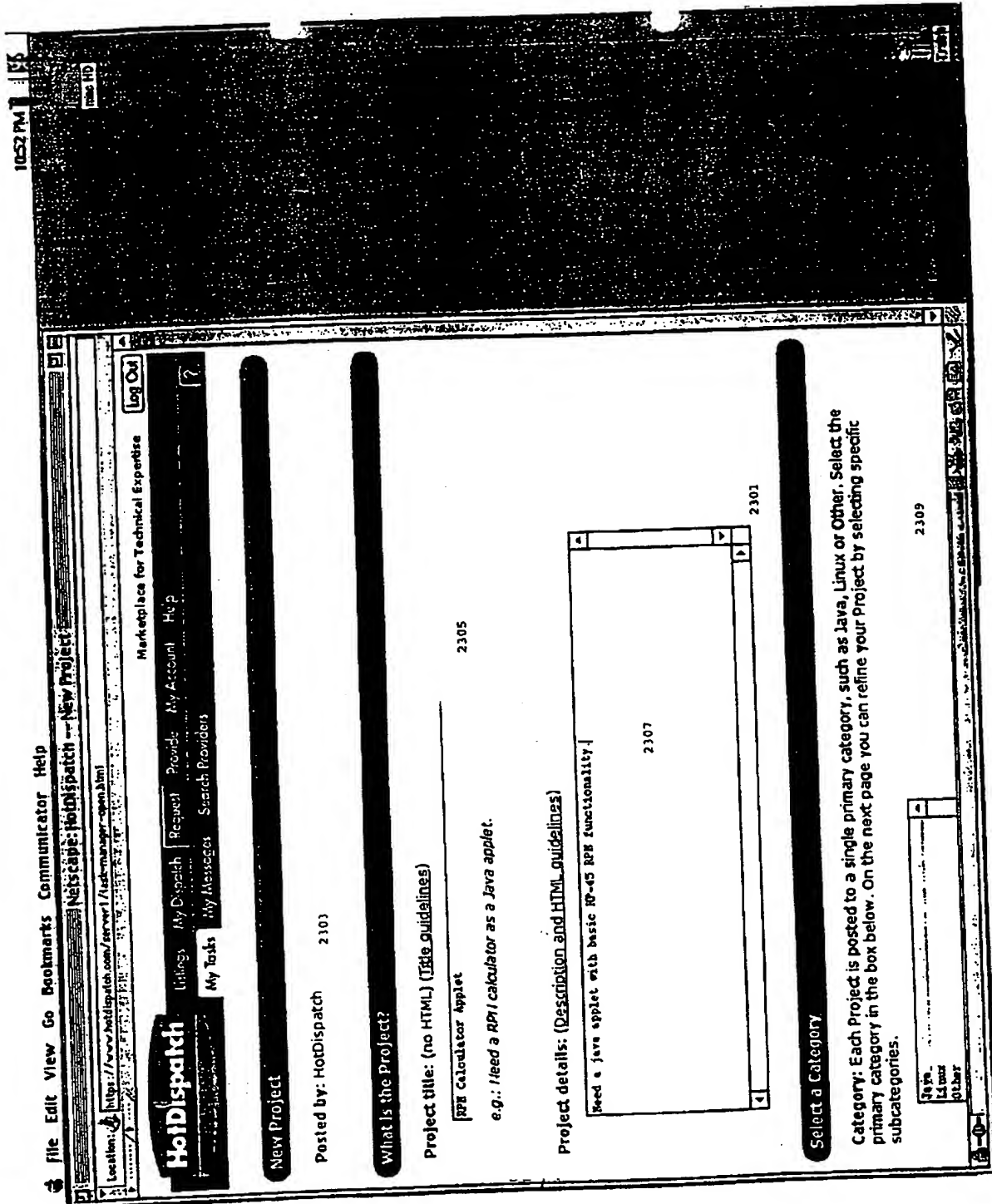
2207

2201

Current profile: HotDispatch

[Legal notice](#) | [Feedback to HotDispatch](#)

Copyright © 1999 HotDispatch, Inc. Use of this website constitutes acceptance of Terms of Service | [Privacy Policy](#)



File Edit View Go Bookmarks Communicator Help

Location: <https://www.hotdispatch.com/server1/task-manager-quick.html>

What is the Price?

First select one of the following pricing options: **2**

☐ Fixed price / nonnegotiable You set the price and only the providers who accept it can be assigned the project.

☐ Fixed price / negotiable You set the starting price, and interested providers can negotiate with you over the final price. Use this pricing scheme if you want to receive offers from multiple providers without having to commit to assigning it to the winner of an auction.

☐ Reverse auction You set the maximum price which becomes the starting price of the auction. Providers can repeatedly bid on your project driving the price down until the auction closes. The project is "assigned" to the provider with the lowest bid.

☒ Sealed bidding Similar to the reverse auction, except a participating provider can only bid once. Bids are not visible to the participating providers. At the end of the auction the **second lowest** bidder is "assigned" the project.

☐ Dutch auction You set a starting low price, which then goes up a fixed dollar amount every given time increment you specify. The first provider who agrees to the current price wins the assignment. While you can start at zero, if your dollar increment is low it might take a long time before the price reaches the level of interest to the providers you seek.

Next Step

Enter pricing information 2401

Legal Notices | Feedback to HotDispatch

Current profile: HotDispatch

Copyright © 1999 HotDispatch, Inc. Use of this website constitutes acceptance of User Agreement | Privacy Policy

10:53 PM

1996

File Edit View Go Bookmarks Communicator Help

<https://www.hotdispatch.com/server/new-task.html>

Primary category:

Software Development / Programming Languages / Java 2503

Subcategories:

☐ Algorithms
 ☒ Java Advanced Imaging
 ☐ Mathematics

☒ Applets
 ☐ Java Foundation Classes
 ☐ Memory Management/GC

☐ AWT
 ☐ Java IOL
 ☐ Multimedia

☐ Beans
 ☐ Java Plugin
 ☐ Networking/Serialization

☐ Bugs and Exceptions
 ☐ Java Server Pages
 ☐ Other

☐ Collections and Enumerations
 ☐ Java Virtual Machine
 ☐ Profiling & Optimization

☐ CORBA
 ☐ JavaHelp
 ☐ RMI

☐ Dates and Formatters
 ☐ JavaMail
 ☐ Security

☐ Design Patterns
 ☐ JDBC
 ☐ Servlets

☐ Input/Output
 ☐ JINI
 ☐ Streams & Strings

☐ Internationalization
 ☐ JIT Compilation
 ☐ Third party libraries

☐ Interoperability
 ☐ JNDI
 ☐ Threads

☐ JAR Files
 ☐ Java 2D
 ☐ XML

☐ Java 3D
 ☐ Language Spec
 ☐

2507

Browser:

None

2501

Java Version:

None

What is the Price?

Sealed bidding: Similar to the auction, except a participating provider can only bid once. Bids are not visible to the participating providers. At the end of the auction the second lowest bidder is "assigned" the task.

Maximum accepted price: \$ 500 2511

This is the highest price (in dollars) you are willing to pay for your task and is the starting price of the bidding. Setting this too low may discourage responses.

Duration of bidding: 1 Day(s) 2513

This is the time period that bidding is permitted on your task.

Important Notes:

1. Your final cost will be the auction's winning price plus the HotDispatch transaction fee of 15%.
2. While your credit card is authorized for the maximum price plus 15%, you are only charged the final price plus the 15%. Your card is authorized when you post the task and charged when the task is closed (you receive and accept the solution).

File Edit View Go Bookmarks Communicator Help

Location: <https://www.hotdispatch.com/server/taskmanager-open.html>

HotDispatch

Marketplace for Technical Expertise

Log Out

My Tools My Messages Search Providers

Request Provide My Account Help

Refresh ?

New Project 2605

New Question 2603

▼ Open task requests (3 tasks) 2607

Date posted	Time left	Status	Task	Categories	Price
11/11 1:23 PM	N/A	Released	question to ganesh	Other	\$30.00 fixed
11/05 4:24 AM	N/A	Solved	first question is just a test	Other	\$30.00 fixed
11/03 8:54 PM	N/A	Solved	Getting Started	Other	\$30.00 fixed

▼ Closed task requests (0 tasks) 2609

Current profile: HotDispatch

2601 Legal Notices | Feedback to HotDispatch

Copyright © 1999 HotDispatch, Inc. Use of this website constitutes acceptance of User Agreement Privacy Policy

File Edit View Go Bookmarks Communicator Help

Log Out

HotDispatch! [Home](#) [My Dispatch](#) [Request](#) [Provide](#) [My Account](#) [Help](#)

[My Tasks](#) [My Messages](#) [Search Providers](#)

Markets for Technical Expertise

Find a Provider

Providers returned will match at least one of the categories selected below.
Or, if you know the provider's name you may enter it in the field at the right.

Search by Name

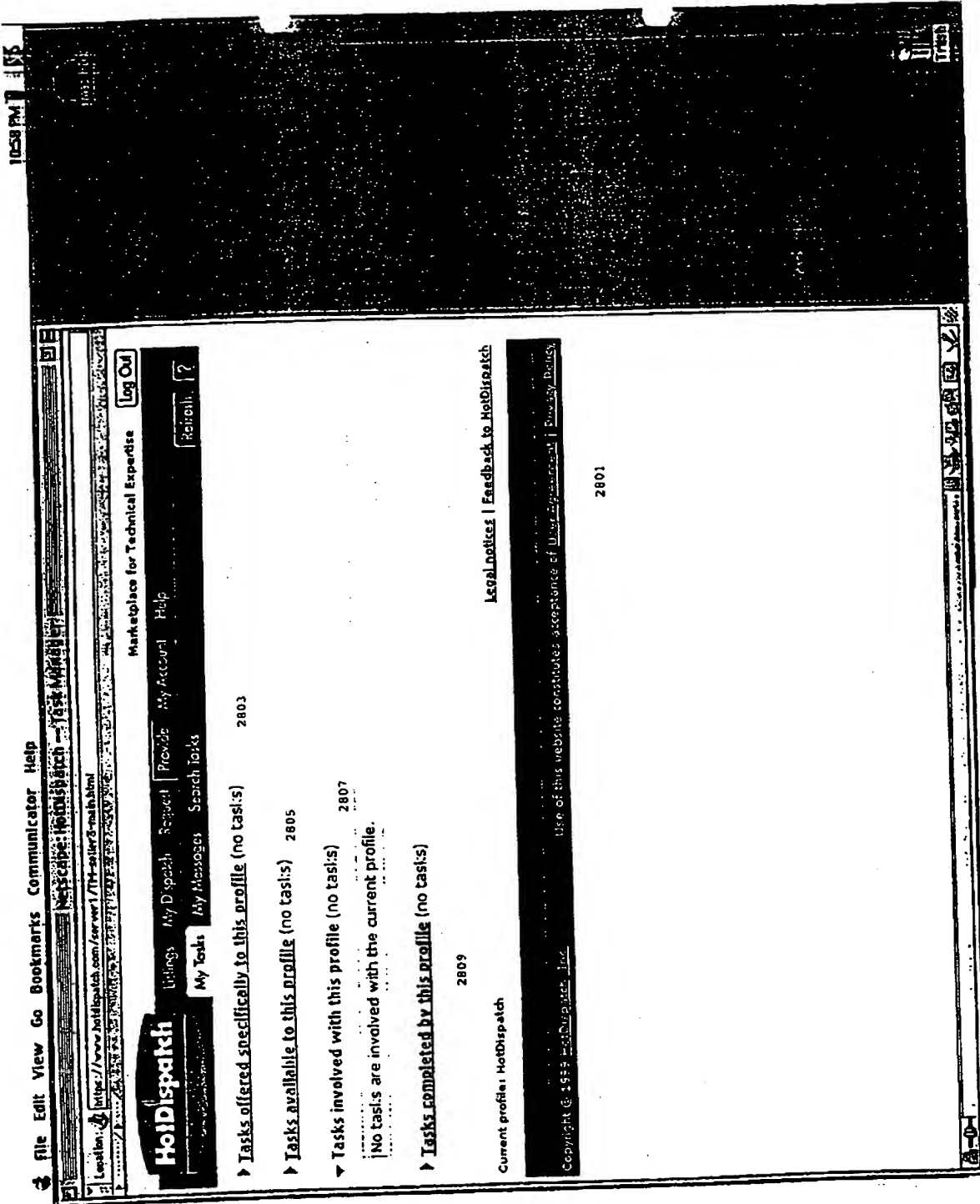
Current category: Java

<input type="checkbox"/> Algorithms (131)	<input type="checkbox"/> IDE: VisualCafe (53)	<input type="checkbox"/> JDBC (724)
<input type="checkbox"/> Applets (771)	<input type="checkbox"/> Input/Output (806)	<input type="checkbox"/> JINI (172)
<input type="checkbox"/> AWT (139)	<input type="checkbox"/> Internationalization (299)	<input type="checkbox"/> JIT Compilation (23) 2705
<input type="checkbox"/> Beans (544)	<input type="checkbox"/> Interoperability (44)	<input type="checkbox"/> JNDI (35)
<input type="checkbox"/> Browser: Internet Explorer (84)	<input type="checkbox"/> JAR Files (99)	<input type="checkbox"/> JMI (271)
<input type="checkbox"/> Browser: Netscape Navigator (85)	<input type="checkbox"/> Java 2D (44)	<input type="checkbox"/> Language Spec (354)
<input type="checkbox"/> Browser: Other (20)	<input type="checkbox"/> Java 3D (26)	<input type="checkbox"/> Mathematics (468)
<input type="checkbox"/> Bugs and Exceptions (601)	<input type="checkbox"/> Java Advanced Imaging (24)	<input type="checkbox"/> Memory Management/GC (65)
<input type="checkbox"/> Collections and Enumerations (561)	<input type="checkbox"/> Java Foundation Classes (619)	<input type="checkbox"/> Multimedia (296)
<input type="checkbox"/> CORBA (67)	<input type="checkbox"/> Java IDL (48)	<input type="checkbox"/> Networking/Serialization (756)
<input type="checkbox"/> Dates and Formatters (472)	<input type="checkbox"/> Java Plug-in (40)	<input type="checkbox"/> Other (344)
<input type="checkbox"/> Design Patterns (516)	<input type="checkbox"/> Java Server Pages (68)	<input type="checkbox"/> Profiling & Optimization (372)
<input type="checkbox"/> IDE: CodeWarrior (14)	<input type="checkbox"/> Java Version: Java 2 (150)	<input type="checkbox"/> RMI (102)
<input type="checkbox"/> IDE: JBuilder (59)	<input type="checkbox"/> Java Version: JDK 1.0 (79)	<input type="checkbox"/> Security (343)
<input type="checkbox"/> IDE: Kawa (22)	<input type="checkbox"/> Java Version: JDK 1.1 (145)	<input type="checkbox"/> Servlets (129)
<input type="checkbox"/> IDE: NetBeans (27)	<input type="checkbox"/> Java Virtual Machine (69)	<input type="checkbox"/> Streams & Strings (759)
<input type="checkbox"/> IDE: Other (24)	<input type="checkbox"/> JavaHelp (28)	<input type="checkbox"/> Third party libraries (319)
<input type="checkbox"/> IDE: Visual J++ (31)	<input type="checkbox"/> JavaMail (41)	<input type="checkbox"/> Threads (135)
<input type="checkbox"/> IDE: VisualAge (46)	<input type="checkbox"/> JavaScript (399)	<input type="checkbox"/> XML (395)

2707 2701

Providers satisfying search criteria (26 profiles) 2709

1 2 3 Next



File Edit View Go Bookmarks Communicator Help

Location: http://www.hotdispatch.com/server/...

HotDispatch

[Listings](#)
[My Dispatch](#)
[Request](#)
[Provide](#)
[My Account](#)
[Help](#)

[My Tasks](#)
[My Messages](#)
[Search Tasks](#)

[Log Out](#)

Marketplace for Technical Expertise

Open task messages (0 messages) 2903

No messages about open tasks are available for the current profile.

Closed task messages (0 messages) 2905

Current profile: HotDispatch

Legal notices | Feedback to HotDispatch

Copyright © 1999 HotDispatch, Inc.

Use of this website constitutes acceptance of our Terms of Service.

File Edit View Go Bookmarks Communicator Help

Location: https://www.hotdispatch.com/.../provider-search-tasks.html

HotDispatch

My Tasks

My Messages

Search Tasks

My Account

Help

HotDispatch

Find a Task

Find a task by its ID: Find Tasks

3005

3007

3003

Browse Tasks by Category

3009

Java

Linux

Other

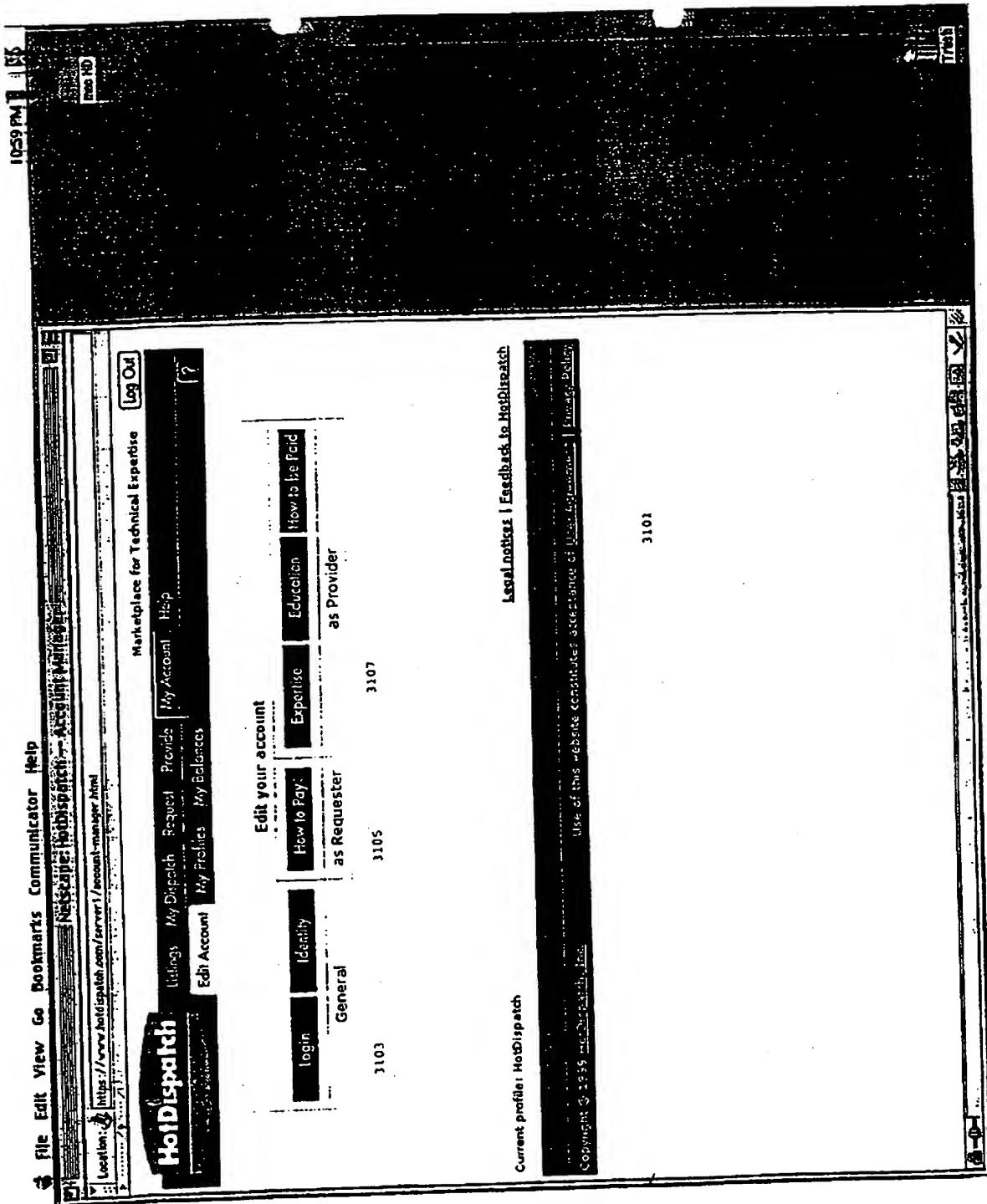
Hot Tasks

Date	Task	Price
11/11/1999	HTTP REFERER as a session variable	\$10.00 (negotiable)
11/10/1999	Java 1.2 Draw & Drop Applet will not run, but does run...	\$25.00 fixed
11/9/1999	I need to close out a html window and open a new window	\$0.00 fixed
11/9/1999	Need help on internationalization	\$0.00 fixed
11/9/1999	How can I make an e-mail a friend a link?	\$0.00 fixed
11/9/1999	JScrollPane problem	\$5.00 fixed
11/9/1999	NoClassDefFoundError after installing JSDK 2.0	\$0.00 fixed
11/9/1999	Little Fast Food Ordering Java Applet	\$0.00 (negotiable)
11/8/1999	Text doesn't display on my screen (in XFree86)	\$0.00 fixed
11/7/1999	Convert GB2312 to UNICODE/UTF-8 and view in IE.	\$0.00 fixed

3011

3001

- 79 -



- 81 -

11:00 PM 1/15

File Edit View Go Bookmarks Communicator Help
HotDispatch

Location: https://www.hotdispatch.com/serve/1/activity-statement.html
Marketplace for Technical Expertise
Log Out

HotDispatch

[Listings](#)
[My Dispatch](#)
[Request](#)
[Provide](#)
[My Account](#)
[Help](#)

[Edit Account](#)
[My Profiles](#)
[My Balances](#)
?

▼ Summary of all the transactions in this account

Additions	\$0.00	Subtractions	\$0.00	Total	3303
-----------	--------	--------------	--------	-------	------

▼ Account activities (0 statements)

No activities are recorded for the current account.

Current profile: HotDispatch

Legal notices | Feedback to HotDispatch

Copyright © 1998 HotDispatch, Inc.

Use of this website constitutes acceptance of our agreement | Privacy Policy

3305

3301

File Edit View Go Bookmarks Communicator Help

Location: <https://www.hotdispatch.com/server/ahet-main.html>

HotDispatch

Marketplace for Technical Expertise

Log Out

My Dispath Request Provide My Account Help

My Tasks My Messages Search Providers Refresh ?

Open task messages (17 messages) 3403 1 2 Next

Action	Date	From	Message
<input type="checkbox"/> Reply	11/11 2:17 PM	ganesh->me	question to ganesh
<input type="checkbox"/> Reply	11/09 12:44 PM	Henry2->me	first question is just a test
<input type="checkbox"/>	11/09 4:38 AM	me->Lemke	first question is just a test
<input type="checkbox"/>	11/09 4:37 AM	me->Joshie	first question is just a test
<input type="checkbox"/>	11/09 4:37 AM	me->dyflet	first question is just a test
<input type="checkbox"/>	11/09 4:36 AM	me->Henry2	first question is just a test
<input type="checkbox"/>	11/08 8:34 PM	System->me	first question is just a test
<input type="checkbox"/>	11/08 6:03 PM	System->me	first question is just a test
<input type="checkbox"/>	11/08 1:09 PM	System->me	first question is just a test
<input type="checkbox"/>	11/07 4:00 AM	System->me	first question is just a test

Deleted Messages

3405

Closed task messages (0 messages)

1 2 Next

3401

HotDispatch

File Edit View Go Bookmarks Communicator Help
[Log Out](#) | [Home](#) | [HotDispatch.com](#) | [New User](#) | [Feedback](#) | [Privacy Policy](#) | [Terms of Service](#) | [Contact Us](#)

Navigation: Home My Dispatch Request Provide My Account Help
My Task My Messages Search Tasks Log Out

Project

Task title:	Computer a 1000 integer prime (last ID: 41752)	
Categories:	Other	3503
Posted by:	HotDispatch (2.5/2.0)	
Posted:	11/17 6:58 AM	
Price:	\$30.00 (negotiable)	
Time left:	N/A	
Description:	computer a 1000 integer prime number	

Clarifications: None

Status: Uninvolved (no solutions)

Request Assignment Send Message 3505

Solutions and Messages

You have not yet become involved with this task.

Current profile hoops 3507

[Legal notice](#) | [Feedback to HotDispatch](#)

Copyright © 1999 HotDispatch.com
 Use of this website constitutes acceptance of terms of agreement | Privacy Policy

File Edit View Go Bookmarks Communicator Help

<https://www.hotdispatch.com/seeover/AnswerIndexProvider.html>

HotDispatch

[Listings](#)
[My Dispatch](#)
[Request](#)
[Provide](#)
[My Account](#)
[Help](#)

[My Tools](#)
[My Messages](#)
[Search Jobs](#)

[Marketplace for Technical Expertise](#)
[Log Out](#)

Offer amount: 33.00 3603

Reply:

I can compute the prime of the first 1000 digit number. see my profile for examples of prior math work.

3605

Submit

Clear Reply

Current profile: hoops

Legal notices | Feedback to HotDispatch

Copyright © 1997 HotDispatch, Inc.

Use of this website constitutes acceptance of User Agreement | Privacy Policy

3601

- 86 -

Marketplace for Technical Expertise

[My Dispatch](#) [Request](#) [Provide](#) [My Account](#) [Help](#)

My Tools
My Messages
Search Providers

Robesh

Project

Task title: Computer a 1000 integer prime (task ID: 41752)

Categories: Other

Posted: 11/17 6:58 AM

price: \$30.00

Time left: N/A

Description:	computer a 1000 integer prime number
---------------------	--------------------------------------

Clarifications: None

Status: Offers Pending

Add Clarification

Solutions and Messages

50000

Assignment Requested

Assignment Requested
3809
The seller's latest offer: \$33.00 (your previous offer:

1118 346
\$30.00

om-Stat. Type

2010-11-14

Dispatch Message

--

patch

1

Journal of Management Education

1900-1901

Current profile: HotDispatch

Legal notices | Feedback to HotDispatch

File Edit View Go Bookmarks Communicator Help

https://www.hotdispatch.com/...

HotDispatch

Marketplace for Technical Expertise

Log Out

Home My Dispatch Request Provide My Account Help

My Tools My Messages Search Providers

To accept this assignment request enter any comments you would like to send to the provider in the box below and then click the submit button

Reply:

Any comments accompanying the bid go here.

3905

3903

Submit

Clear Reply

3907

Current profile: HotDispatch

Legal notices | Feedback to HotDispatch

Copyright © 1999 HotDispatch.com Use of this website constitutes acceptance of these terms and conditions

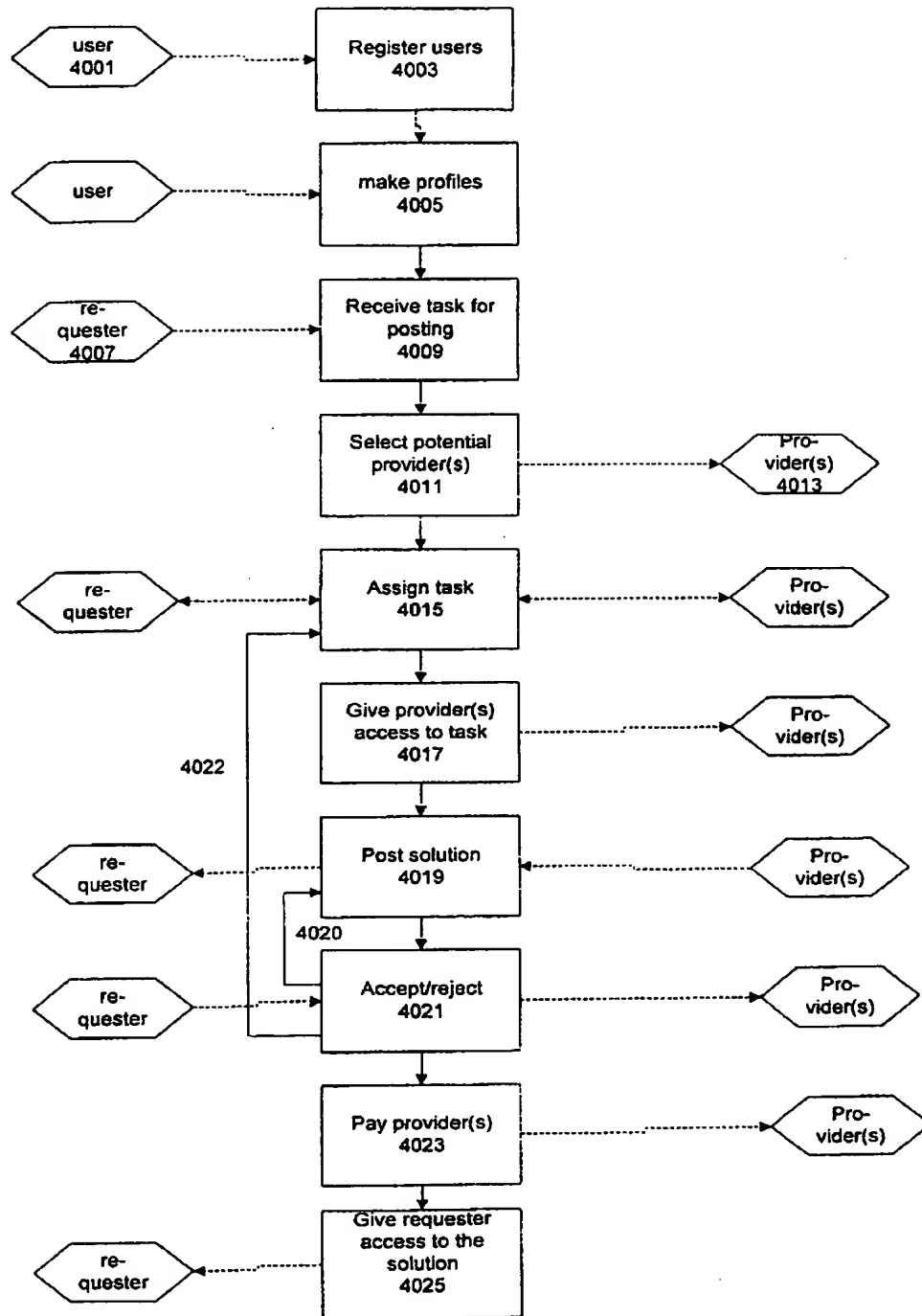


FIG. 40

4001

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US99/27270

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) : G06F 17/30

US CL : 705/27

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 705/10, 20, 26-27; 707/10

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

WEST

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A,P	US 6,029,195 A (HERZ) 22 FEBRUARY 2000, SEE ABSTRACT, COLUMN 64, LINE 2 TO COLUMN 70, LINE 17.	1-2
A	²⁵ 6,006,251 A (TOYOUCHI ET AL.) 21 DECEMBER 1999, SEE ABSTRACT AND COLUMN 17, LINE 20 TO COLUMN 19, LINE 43.	1-2
A,P	US 5,950,173 A (PERKOWSKI) 07 SEPTEMBER 1999, SEE ABSTRACT.	1-2
A	US 4,992,940 A (DWORKIN) 12 FEBRUARY 1991, SEE ABSTRACT AND FIGURE 6.	1-2

☐ Further documents are listed in the continuation of Box C.
 ☐ See patent family annex.

* Special categories of cited documents	*T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
A document defining the general state of the art which is not considered to be of particular relevance	*X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
E earlier document published on or after the international filing date	*Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
L document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	*G* document member of the same patent family
O document referring to an oral disclosure, use, exhibition or other means	
P document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search

25 MARCH 2000

Date of mailing of the international search report

14 APR 2000

 Name and mailing address of the ISA/US
 Commissioner of Patents and Trademarks
 Box PCT
 Washington, D.C. 20231
 Facsimile No. (703) 305-3230

Authorized officer

FRANTZY POINYIL

Telephone No.

James R. Matthews
(703) 305-9775

Form PCT/ISA/210 (second sheet) (July 1998)*

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.